EFFICIENT COMPUTATIONAL TECHNIQUES FOR HIGH DIMENSIONAL STOCHASTIC MODELING

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Jiang Wan

August 2013

EFFICIENT COMPUTATIONAL TECHNIQUES FOR HIGH DIMENSIONAL STOCHASTIC MODELING

Jiang Wan, Ph.D.

Cornell University 2013

Modeling of physical systems in the presence of uncertainties is critical in many respects. Therefore it is necessary to quantitatively characterize these uncertainties. There are two major types of problems with respect to uncertainty quantification: inverse problems and forward problems. In inverse problems, it is essential to estimate the uncertainties arising from limited observation data. In forward problems, the main objective is to understand how input uncertainties propagate and how they affect model responses. In spite of tremendous progress made in the past few decades, the problems arising from high-dimensional input remain a long-standing challenge. The focus of this thesis is developing an efficient computational framework to overcome the curse of dimensionality in both inverse and forward problems.

For inverse problems with high-dimensional input, we develop a Bayesian computational framework in which the input field is discretized using a sparse grid and represented by local basis functions associated with the collocation points. Based on the hierarchical property of sparse grids, a sequence of hierarchical Bayesian models from coarse to fine scales is proposed. The sparse grid also provides an efficient way of finding an optimal choice of basis functions to approximate the spatially varying input, which leads to an adaptive refinement strategy. As a result, it reduces the dimensionality of the inverse problem and the computational cost of Bayesian inference. This Bayesian computational

approach is nonparametric and thus is applicable to various spatially varying parameter estimation problems.

For forward problems with high-dimensional input, probabilistic graphical models, which have been extensively used in machine learning and information science, are employed to approximate the high-dimensional joint probability density functions that exist in uncertainty quantification. We combine the graphical models and a popular model reduction technique, Karhunen-Loève expansion, to construct accurate stochastic input model for non-Gaussian random fields. Furthermore, we develop a probabilistic graphical model based methodology for uncertainty quantification in the presence of both high-dimensional stochastic input and multiple scales. In this framework, the stochastic input and model responses are treated as random variables. Their relationships are modeled by graphical models which give explicit factorization of the high-dimensional joint probability distribution. In this way, an efficient inference algorithm, belief propagation, is applied to infer the statistics of model responses directly on the graph without involving sampling-based methods and expensive deterministic solvers.

BIOGRAPHICAL SKETCH

The authors was born in Anhui, China. After completing his high school education from Wuwei Middle School, the author was admitted into the Bachelor's program in Mechanical Engineering at Tsinghua University in Beijing, China. The author entered the doctoral program at the Sibley School of Mechanical and Aerospace Engineering, Cornell University and was awarded a special Master's degree in 2012.

This thesis is dedicated to my parents Daixiang Wan and Aihua Xu for their constant support and encouragement during my school years.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my advisor, Professor Nicholas Zabaras, for his continuous support and guidance of my Ph.D study and research, for his great patience, motivation and enthusiasm. His immense knowledge and insightful suggestions helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study. I would also like to thank Professors Christopher Earls and Derek Warner for serving on my special committee and for their encouragement and suggestions during the course of this work. Their kindly helps are precious to me.

This research was supported by an OSD/AFOSR MURI09 award on uncertainty quantification, the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research and the Computational Mathematics program of the National Science Foundation (NSF) (award DMS-0809062 and DMS-1214282). This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. Additional computing resources were provided by the NSF through TeraGrid resources provided by NCSA under grant number TG-DMS090007. I would like to thank the Sibley School of Mechanical and Aerospace Engineering for having supported me through a teaching assistantship for part of my study at Cornell. Finally, I would like to thank fellow MPDC members and other friends for their support during my days at Cornell.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Physical systems generally have inherent randomness which could result from uncertainties in boundary or initial conditions, material heterogeneities and so on. It is therefore necessary to include parameters characterizing these uncertainties into the model system. Hence, stochastic ordinary/partial differential equations (SODEs/SPDEs) are constructed for uncertainty quantification. There are two major types of problems with respect to uncertainty quantification: inverse problems and forward problems. In inverse problems, one converts observed measurements (model responses) into information about the input. In the context of uncertainty quantification, a Bayesian framework is often desirable as it efficiently captures the statistical properties of the input. In forward problems, one generally studies the propagation of uncertainties across multiple scales from the stochastic input space to the response space. In practice, many important problems in both categories are related to modeling in high-dimensional spaces. For example, the heterogeneity of subsurface in ground water transport can be represented by multiscale fluctuations in the permeability of the media. The multiscale features of these problems can result in high-dimensional stochastic spaces which make it difficult to conduct an efficient analysis. Over the past few decades, there have been many studies on high-dimensional stochastic modeling. However, the curse of dimensionality remains a challenging issue. The goal of this thesis is to develop new computational techniques for both inverse and forward problems towards improving the efficiency of analysis of complex systems with high-dimensional input.

An important category of inverse problems is the identification of spatially varying parameters using indirect data. A typical example is the permeability estimation of the aquifer from flow data. The measurement error and inadequacy of models for complicated physical phenomena can reduce the accuracy of the estimation [59, 45]. The deterministic approaches address these problems based on exact matching or least squares optimization without quantifying the uncertainty of the solution [84]. Other alternative approaches based on the spectral stochastic method or Bayesian inference [40, 85, 12, 93, 38] take into account the statistical nature of inverse problems and provide full probabilistic description of the computed fields. In Bayesian inference of spatially varying parameters, finite element techniques are often used to discretize the unknown field [49, 56]. Standard models for spatial data, such as Markov random field (MRF) or Gaussian process (GP), are then used to model the spatially varying parameters [49, 26, 56, 45, 5]. To increase the flexibility of the model, the process convolution approach is used as an alternative [45, 56]. By convolving white noise with a smoothing kernel, the unknown parameter field is approximated as a superposition of kernel-type functions centered at various locations. The inverse problem is then transformed to one that infers the coefficients of the expansion. However, these methods are based on some assumptions about the spatial correlation and their performances deteriorate with the increase of complexity, especially the dimensionality of the spatial fields. This challenge motivated us to develop a new way of modeling the unknown field of spatially varying parameters with hierarchical representation of the parameter field based on sparse grid interpolation. The sparse grid collocation (SGC) method uses the Smolyak algorithm to construct an interpolation of the target function with hierarchical grids and basis functions [79, 11]. The collocation points (nodes) are

selected in a nested fashion to obtain many recurring points and basis functions with increasing sparse grid levels [42, 54, 11]. In other words, the basis functions are constructed on multiple scales, which can lead to an adaptive refinement strategy for optimal choice of collocation points [54]. In this way, rapid changes in the spatial field can be effectively captured and an optimal representation of the spatially varying parameter with minimum requirement of collocation points is achieved.

For uncertainty quantification in forward problems, most numerical methods solving SODEs/SPDEs are based on quantitative characterization of the stochastic input. Hence, it is essential to construct a probabilistic model of the random input from available information. The most common choice for this purpose is the Karhunen-Loève (KL) expansion which represents a random field in terms of a linear combination of deterministic basis functions and orthonormal random variables called KL random variables. A finite number of expansion terms are then retained to optimally reduce the number of random variables needed to characterize the random field. Its nonlinear variant carries out the same idea but deals with high-dimensional input in a feature space. In a word, these methods project high-dimensional stochastic input into a lower-dimensional space. In this way, samples of the random field can be generated from finite dominant random variables. Generally, these model reduction techniques are implemented numerically based on limited experimental data. Since the analytic expression of the joint probability of random variables in the reduced space is intractable, it is desirable to construct a probabilistic model of the KL random variables from data. The polynomial chaos (PC) expansions are most commonly used for this purpose. While the joint distribution could be nonstandard, PC expansions represent them in terms of specific standard

random variables for computational expedience. For example, when Hermite polynomials are applied, the expansion is a function of independent Gaussian random variables. The PC coefficients can be evaluated by Galerkin projections due to the orthogonality of the polynomials [27]. In this framework, a main challenge lies in modeling the joint probability density of random variables projected in a lower-dimensional space, e.g. the KL expansion random variables. For a Gaussian random field, these variables are mutually independent and it is straightforward to decompose the multivariate joint probability to the product of 1D marginal probability density functions. Various methods, parametric or nonparametric, can be employed to construct probabilistic models of these random variables individually from data. However, non-Gaussian random fields are more realistic in many instances, in which cases the joint probabilities take more complex forms and thus are more difficult to estimate accurately from limited data. As it is not realistic to construct a numerical model for an arbitrary random vector based on its entire family of joint probabilities, most common approaches focus on reduced objectives [68]. A typical one is to find a model that has the same one-dimensional marginal probability distributions, which implies that the random variables are mutually independent [73]. This is easy to implement, but obviously, the joint probability of random variables of non-Gaussian random fields cannot be accurately captured in this way. As alternative, several approaches have been proposed to relax the assumption of independence among random variables. One of them is the Rosenblatt transformation [74], which is used to compute the PC coefficients based on sequential conditional distributions of random variables, although it does not explicitly explore the dependencies among these variables [77]. However, by ordering the target random variables in different ways, the Rosenblatt transformation is not

unique and can give different results. Another drawback of this method is that the multivariate joint probability is still estimated in the whole parameter space. In [73], the authors derived the joint probability of random variables in terms of PC coefficients. The likelihood function is approximated by the product of one-dimensional marginal likelihood functions of the target random variables. Then the maximum likelihood method is used to estimate the PC coefficients from observation data. The approximation of the likelihood function improves the computational efficiency at the cost of accuracy. Moreover, it also does not guarantee a unique PC expansion. In [80], a variational method is developed to approximate the multivariate joint probability based on the maximum entropy principle. Given observation data, the joint moments of the target random variables at different orders are obtained numerically and serve as constraints to maximize the entropy function. The dependencies among these random variables are characterized by such joint statistics. However, the number of realizations required to acquire accurate high-order statistics is usually quite large. Given the order of constraint statistics, the number of unknown parameters in the approximated joint PDF increases exponentially with the number of random variables. In [92], the problem of modeling the joint probability is bypassed by postprocessing. The PC expansion is still constructed with the assumption of independent random variables, but two post-processing procedures are proposed on the general polynomial chaos (gPC) solution to get correct statistics that are consistent with the joint PDF of KL expansion random variables. In [68], a multivariate joint probability is estimated based on marginal distributions and a copula function which can represent the dependency information between random variables.

The approximation of the joint probability of KL random variables needs to be made by taking balance between computational expedience and accuracy. While many approaches in previous studies may have good performance under low dimensionality, a relatively large number of retained terms in a truncated KL expansion can lead to great computational challenges. Inspired by the conventional Rosenblatt transformation which is based on sequential conditional distributions, an idea of factorizing the multivariate joint probability into low dimensional conditional distributions comes out. To this end, the Bayesian network (BN) framework is proposed in this thesis. A BN is a directed acyclic graphical model that encodes the joint probability of a set of random variables [8]. The graphical model comprises of nodes, each of which denotes a single random variable, and directed edges that link the nodes. The structure of the network expresses the probabilistic relationships between these nodes. Given observations of the random variables, many BN structure learning algorithms have been developed in recent years to find a probabilistic model of the joint probability that is consistent with the given data [8]. This approach has the advantage that it considers a set of local distributions and does not require to model directly the global distribution. Another advantage is that learning algorithms are better suited in addressing the curse of dimensionality.

The other challenge in forward problems is related to solving multiscale SPDEs. The most celebrated method is the Monte Carlo (MC) method. As a sampling method, the deterministic solver is called for each realization of the stochastic input for one to obtain the statistics of the solution. The convergence rate does not depend on the dimension of the parameter space, but is of order $\mathcal{O}(n^{-1/2})$ with n realizations. To accelerate convergence, quasi Monte Carlo methods and several efficient sampling techniques have been developed as al-

ternatives. Another group of methods refers to nonsampling approaches, typically perturbation algorithms based on a series representation of the stochastic solution. These methods are limited to small fluctuations and low-order statistics of the solution. Recently, many research efforts have been devoted to the study of schemes based on spectral representation of the stochastic solution [66]. For example, the well-established stochastic Galerkin method approximates the solution in a multivariate polynomial space or in anisotropic tensor product polynomial spaces [32]. Stochastic collocation methods using sparse grids based on the Smolyak algorithm [79] have a weaker dependence on the dimensionality of the problem and recently have been applied extensively to various uncertainty quantification problems [96, 95, 65, 55, 57].

In spite of the tremendous progress in solving SPDEs, the curse of dimensionality remains even after using model reduction techniques. For conventional stochastic Galerkin methods, the computational cost depends on the number of expansion terms which grows exponentially as a function of the dimensionality of the stochastic input space [16]. Stochastic collocation methods can achieve fast convergence rate by taking advantage of multidimensional polynomial interpolation. However, the number of collocation points required to achieve sufficient accuracy increases exponentially for high-dimensional problems [58]. Therefore, many efforts have been devoted to stochastic methods that deal with high-dimensions. In [55], an adaptive sparse grid collocation (ASGC) method is proposed such that the collocation points are selected automatically based on the smoothness of the stochastic domain as detected by the magnitude of the hierarchical surpluses. The ASGC can successfully solve stochastic elliptic problems up to 100 dimensions when not all stochastic dimensions are equally important [57]. However, the convergence rate deteriorates

even for problems with moderate input dimensionality when all stochastic dimensions are equally weighted. Another direction is to decompose the original problem into sub-problems with low-dimensional input. A typical example are the high-dimensional model representation (HDMR) techniques that capture the high-dimensional relationships between input and output model variables and generate a collection of low-dimensional sub-problems in stochastic space [50]. Recent progress in HDMR can be found in [57, 99]. In [16], a low-rank separated representation of the solution to SPDEs with high-dimensional inputs is obtained using an alternating least-squares approach.

More challenges arise when multiscale phenomena are taken into account in high-dimensional stochastic problems. In such cases, information across scales contains a certain level of uncertainty but assessment of uncertainty propagation often leads to large computational cost. Let us take, for example, fluid flow through porous media occurring from large geological scales down to microscopic scales. Full-scale spatial and temporal resolution simulations may require significant computational resources. Since the sample-based stochastic methods mentioned above, such as MC, ASGC, and HDMR, call the deterministic solvers repetitively, efficient solvers for multiscale partial differential equations are of great importance in reducing overall computational cost. For this purpose, computational techniques, such as the multiscale finite element (MsFEM) method [34, 35, 25], variational multiscale (VMS) method [36, 37], the heterogeneous multiscale (HMM) method as well as their variants [58, 21, 22] and multigrid methods [91, 24], have been developed to solve a coarse-scale problem that captures fine-scale effects without resolving all the fine-scale features.

So far, many efforts for solving multiscale stochastic problems focus on decoupling multiscale deterministic solvers from stochastic approaches. We propose here a new scheme to quantify the uncertainties propagated in multiscale systems based on a probabilistic model of SPDE solutions. Inference problems can be solved directly on this probabilistic model without sampling-based methods or calling expensive deterministic solvers. The stochastic input and model responses are all treated as random variables. However, conventional regression models are inefficient or even impractical to represent their relationships when the stochastic input is in a high-dimensional space. This curse of dimensionality can be overcome by utilizing probabilistic undirected graphical models which have been intensively studied and widely used in machine learning and Bayesian statistics for multivariate statistical modeling [94, 86, 9]. Similar with BN, an undirected graphical model also consists of a collection of nodes and edges except that the edges are not orientated (possibly because the causality between random variables are implicit or not available). By combining both graph theory and probability theory, the complicated relationships between all variables can be modeled explicitly and the resulting graph expresses a decomposition of a joint distribution as a product of functions of subsets of variables. Given the graph-based probabilistic model of model responses, efficient algorithms for inference on graphical models can be directly applied. If we treat the stochastic input as observed variables, the probabilistic model becomes a conditional distribution of model responses on input variables, which leads to a surrogate model. The predictions of model responses can be evaluated by inference algorithms on the graphical model associated with this surrogate model. If the stochastic input field also has an explicit graphical representation, we can directly estimate the marginal distributions of unobserved variables in

the graphical model by integrating out all the other variables with an efficient algorithm.

The organization of this thesis is as follows: In Chapter 2, a Bayesian computational approach based on a hierarchical representation of parameter space is proposed to solve inverse problems with high-dimensional input. In Chapter 3, we construct a stochastic input model in a high-dimensional space with probabilistic graphical models. In Chapter 4, a probabilistic graphical model based methodology is developed to efficiently perform uncertainty quantification in the presence of both stochastic input and multiple scales. Finally, conclusions of this thesis and suggestions for future research are summarized in Chapter 5.

CHAPTER 2

SOLVING MULTISCALE INVERSE PROBLEMS: A BAYESIAN APPROACH BASED ON THE SEQUENTIAL MONTE CARLO METHOD

In this chapter, we develop a Bayesian computational approach to estimate spatially varying parameters. Most content of this chapter is from the work in [88]. The sparse grid collocation method is adopted to parameterize the spatial field. Based on a hierarchically structured sparse grid, a multiscale representation of the spatial field is constructed. An adaptive refinement strategy is then used for computing the spatially varying parameter. A sequential Monte Carlo sampler is used to explore the posterior distributions defined on multiple scales. The SMC sampling is directly parallelizable and is superior to conventional MCMC methods for multi-modal target distributions. The samples obtained at coarser levels of resolution are used to provide prior information for the estimation at finer levels. This Bayesian computational approach is rather general and applicable to various spatially varying parameter estimation problems.

2.1 Parameterization of the unknown parameter field

The spatially varying parameter of a physical system belongs to an infinite dimensional space. In a Bayesian framework, it is usually reduced to a finite space and the inference is performed on a finite set of random variables. A simple way of implementing this task is to discretize the spatial domain into finite elements. The value of the spatially varying parameter is assumed constant within each element [49]. However, if improper resolution of discretization is selected, either overfitting or a waste of computational resources takes place [45]. To in-

crease the flexibility of the model, several researchers consider a basis functions approach, such as in the truncated Karhumen-Loève expansion (KLE) or using Gaussian kernels, to represent the unknown parameter field [56, 45, 17, 18]. However, these methods might require prior knowledge of the correlation or covariance functions of the stochastic process. In addition, an optimal choice of the basis could be a difficult task. Although the number of basis functions is not fixed and treated as a random variable in the trans-dimensional MCMC method, it is difficult to alter the dimension significantly while ensuring a reasonable acceptance ratio [28, 72].

In this work, we propose a hierarchical basis representation of the spatially varying parameter based on the sparse grid interpolation method. The basic idea is to have a hierarchical structure of representation that ranges from coarse to fine scales of discretization. Thus we can perform sequential Bayesian estimation from coarse scale with a few number of collocation points until an optimal choice of collocation points and basis functions is achieved. The method is still based on a finite element discretization of the spatial domain for the purpose of solving the underlying differential equations, but the unknown parameter field is approximated using interpolating functions on a set of collocation points.

The sparse grid method is a special discretization technique which constructs hierarchical interpolation based on the Smolyak algorithm [11, 42, 54]. The standard sparse grid is defined with collocation points and basis functions completely fixed at different levels of resolution. Given a sparse grid, the unknown function for the spatially varying parameter can be approximated using basis functions associated with the grid. For example, consider a parameter θ in 1D, the function $\theta = f(x)$ is approximated by the nodal basis of interpolation

level 2 as (Fig. 2.1)

$$\theta = f(x) \approx w_1^1 a_1^1 + w_1^2 a_1^2 + w_2^2 a_2^2. \tag{2.1}$$

On level 3, it is approximated as

$$\theta = f(x) \approx w_1^1 a_1^1 + w_1^2 a_1^2 + w_2^2 a_2^2 + w_1^3 a_1^3 + w_2^3 a_2^3, \tag{2.2}$$

by hierarchically adding two basis functions.

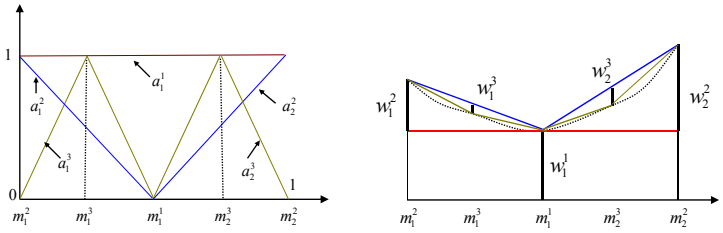

Figure 2.1: Hierarchical basis functions a_j^i with the support nodes (left) and the hierarchical surpluses (right). The surplus w_j^i is defined as the difference between the function value computed at a newly added point on the current sparse grid and the interpolated value at this point from the previous interpolation level.

The sparse grid provides several advantages in Bayesian inference of spatially varying parameters. First, the accuracy of the interpolation is increased without discarding previous results. The reusability of the collocation points and basis functions enables a sequential estimation of the surpluses from coarse scales. One does not have to set up a fine grid and estimate a large number of surpluses simultaneously. Furthermore, the definition of the surplus and the nested fashion of the collocation points potentially enable an adaptive refinement of the grid. The collocation points of the adaptive sparse grid are case determined and are only a subset of the nodes of the standard sparse grid at the same level, which further reduces the dimensionality of the inverse problem. Detailed discussion about these topics will be given in later sections as part of our discussion of the hierarchical Bayesian model.

2.2 Bayesian inference

In the last section, the spatially varying parameter is parameterized by the sparse grid method. In the Bayesian framework, the unknown surpluses are treated as random variables and inferred from the observation data. In this section, a complete one-scale Bayesian model is developed on a pre-determined sparse grid. The sequential Monte Carlo sampler is utilized for an efficient exploration of the posterior state space.

2.2.1 Bayesian formulation

Consider a generalized forward problem defined as

$$\boldsymbol{d} \approx \boldsymbol{F}(\theta), \tag{2.3}$$

where $\boldsymbol{d}$ denotes the observation data and θ denotes the model parameter which is considered as a random variable or a random vector. Based on Bayes' theorem, the posterior probability density for θ is

$$p(\theta|\boldsymbol{d}) \propto p(\boldsymbol{d}|\theta)p(\theta). \tag{2.4}$$

Here, $p(\boldsymbol{d}|\theta)$ is the likelihood distribution and $p(\theta)$ is the prior probability density. The forward solver $\boldsymbol{F}$ gives predictions by solving parameterized partial differential equations (PDEs) with numerical methods such as the finite element method (FEM).

Prior specification

When the model parameter θ is a spatially varying parameter, the forward problem can be reformulated as

$$\boldsymbol{d} \approx \boldsymbol{F}(f(\boldsymbol{x})), \tag{2.5}$$

where $\boldsymbol{x}$ refers to the location and $f(\boldsymbol{x})$ is the specific field of the spatially varying parameter, i.e. $\theta = f(\boldsymbol{x})$. On a pre-determined sparse grid at level q, the parameter field can be represented as a weighted sum of the basis functions for all collocation points in the sparse grid from level 0 to q due to the hierarchical structure of the grid, that is

$$\theta \approx \theta_q = f_q(\boldsymbol{x}) \quad = \quad \sum_{i=0}^{q} \sum_{j=1}^{k_i} w_j^i \cdot a_j^i(\boldsymbol{x}). \tag{2.6}$$

Hence, our problem is reduced to the inference of the surpluses, w_j^i. A simple non-informative prior for the surpluses is the multivariate Gaussian distribution which assumes the surpluses are identically independently distributed as:

$$\boldsymbol{w}_q | \sigma_{w,q} \sim \mathcal{N}(0, \sigma_{w,q}^2 \mathbf{I}_k), \tag{2.7}$$

where $\boldsymbol{w}_q$ is the vector of all surpluses up to level q for θ_q and $\mathbf{I}_k$ is an identity matrix. Suppose the hyperparameter $\sigma_{w,q}^{-2} \sim \Gamma(\alpha_0, \beta_0)$, we can premarginalize the variance from Eq. (2.7) and obtain the prior

$$p(\boldsymbol{w}_q) \propto \frac{\Gamma(\alpha_0 + \frac{k}{2})}{(\beta_0 + \frac{1}{2}\|\boldsymbol{w}_q\|_2^2)^{\alpha_0 + \frac{k}{2}}}, \tag{2.8}$$

where k is the length of the vector $\boldsymbol{w}_q$.

Likelihood

To evaluate the likelihood, the discrepancy between the forward solver prediction and observation data should be estimated and modeled. Two primary

sources of error are considered in the modeling. One is the measurement noise ζ, which is generally assumed to be independent additive Gaussian random error with mean zero, i.e.

$$\zeta_i \sim \mathcal{N}(0, \sigma_\zeta^2), \tag{2.9}$$

where ζ_i is an element of the vector ζ. We postulate the following relationship

$$\boldsymbol{d} = \boldsymbol{F}(\theta) + \zeta, \tag{2.10}$$

i.e. the observation is obtained from accurate forward solver predictions plus measurement noise.

The other source is the model error δ, or referred to as the approximation error, which results from the inadequacy of the forward model to represent the real physical process [41, 33]. To minimize this effect, the forward solver operates at a fixed fine scale in this work. We restrict our focus on the part of model error that results from the parameterization of the spatially varying parameters.

With a sparse grid representation of the spatially varying parameter θ, the accurate parameter field $f(\boldsymbol{x})$ is transformed into a reduced approximative model described by Eq. (2.6). This approximation error will propagate in the forward solver and result in inaccurate predictions. Thus, the observation is formulated as

$$\begin{aligned} \boldsymbol{d} &= \boldsymbol{F}(\theta) + \zeta \\ &= \boldsymbol{F}_q(\theta_q) + (\boldsymbol{F}(\theta) - \boldsymbol{F}_q(\theta_q)) + \zeta \\ &= \boldsymbol{F}_q(\theta_q) + \delta + \zeta, \end{aligned} \tag{2.11}$$

where q is the level of the sparse grid, $\boldsymbol{F}_q(\theta_q)$ refers to the forward solver using the approximate model θ_q for the spatially varying parameter θ. The model error is defined as $\delta = \boldsymbol{F}(\theta) - \boldsymbol{F}_q(\theta_q)$, the discrepancy of forward solver predictions

using precise spatially varying parameters and using an approximate model as in Eq. (2.6).

In Bayesian inference, the model error δ can be treated as a random field. A Markov random field is employed as in [26]. The basic idea is that the model error at a particular location is correlated with the model errors at neighboring locations. Suppose a random vector $e = \delta + \zeta$, we assume,

$$e \sim \mathcal{N}(\delta_0, \Sigma_e), \tag{2.12}$$

where $\delta_0 = \delta_q \mathbf{1}$ and the covariance matrix Σ_e is exponential formulated as

$$\Sigma_e = \sigma_\zeta^2 \mathbf{I} + \kappa H(\phi). \tag{2.13}$$

The parameter σ_ζ^2 represents the variance of the measurement error. $H(\phi) = \{H_{ij}(\phi)\}$ where $H_{ij}(\phi) = \exp(-\frac{||s_i - s_j||}{\phi})$ and $||s_i - s_j||$ is the Euclidean distance between locations s_i and s_j. This is a measure of the magnitude of spatial dependence. κ and ϕ indicate the scale and the range of the spatial dependence, respectively [29]. Thus, the likelihood for general data modeling is

$$\boldsymbol{d}|\boldsymbol{w}_q, \delta_q, \Sigma_e \sim \mathcal{N}(\boldsymbol{F}_q(\boldsymbol{w}_q) + \delta_q \mathbf{1}, \Sigma_e). \tag{2.14}$$

To further reduce the dimensionality of the problem as well as the complexity of computation, we also adopt a much simplified model. We simply assume that δ is a random field (vector) with independent elements subject to a multivariate Gaussian distribution:

$$\delta \sim \mathcal{N}(\delta_0, \sigma_\delta^2 \mathbf{I}). \tag{2.15}$$

Since $\zeta \sim \mathcal{N}(0, \sigma_\zeta^2 \mathbf{I})$ according to Eq. (2.9), e is also subject to a Gaussian distribution $e \sim \mathcal{N}(\delta_0, \Sigma_e)$ where $\Sigma_e = (\sigma_\delta^2 + \sigma_\zeta^2)\mathbf{I}$. Thus, the likelihood is simply

$$\boldsymbol{d}|\boldsymbol{w}_q, \delta_q, \sigma_e \sim \mathcal{N}(\boldsymbol{F}_q(\boldsymbol{w}_q) + \delta_q \mathbf{1}, \sigma_e^2 \mathbf{I}), \tag{2.16}$$

where $\sigma_e^2 = \sigma_\delta^2 + \sigma_\zeta^2$. In this model, a gamma distribution $\Gamma(\alpha_1, \beta_1)$ is taken for σ_e^{-2}. By premarginalization, the unknown variance can be integrated out. The likelihood function is simply

$$p(\boldsymbol{d}|\boldsymbol{w}_q, \delta_q) \propto \frac{\Gamma(\alpha_1 + \frac{m}{2})}{(\beta_1 + \frac{1}{2}\|\boldsymbol{d} - \boldsymbol{F}_q(\boldsymbol{w}_q) - \delta_q \mathbf{1}\|_2^2)^{\alpha_1 + \frac{m}{2}}}, \tag{2.17}$$

where m is the number of observation data. In the following sections, we refer to the former model as Model I and to the latter one as Model II.

Complete Bayesian model

Consider a pre-determined sparse grid at level q and let $\psi_q = \{\boldsymbol{w}_q, \delta_q, \sigma_\zeta, \kappa, \phi\}$ denote the vector containing all unknown parameters of the Bayesian model on this grid. The prior for the hyperparameter δ_q is taken to be: $\delta_q \sim \mathcal{N}(0, \sigma_{\delta_q})$. A gamma distribution $\Gamma(\alpha_{\sigma_\zeta}, \beta_{\sigma_\zeta})$ is chosen as the prior for σ_ζ^{-2}. Since $\kappa > 0, \phi > 0$, two gamma distributions $\Gamma(\alpha_\kappa, \beta_\kappa), \Gamma(\alpha_\phi, \beta_\phi)$ are adopted as priors of κ^{-1} and ϕ^{-1}, respectively. Combining the priors for surpluses (Eq. (2.8)) and hyperparameters and the likelihood from Eq. (2.14), the posterior distribution for Model I is

$$\pi_q(\psi_q) = p(\psi_q|\boldsymbol{d}) \propto p(\boldsymbol{d}|\boldsymbol{w}_q, \delta_q) p(\boldsymbol{w}_q) p(\delta_q) p(\sigma_\zeta) p(\kappa) p(\phi). \tag{2.18}$$

The posterior distribution for Model II with the likelihood from Eq. (2.17) is similar except that the hyperparameters $\sigma_\zeta, \kappa, \phi$ are removed.

2.2.2 Exploring the posterior state space

The Bayesian posterior distribution derived in the previous section is in general analytically intractable. Standard MCMC methods, e.g. Metropolis-Hastings

(MH) sampler and Gibbs sampler, have been extensively used for such problems and their versatility and power have been proved in practical applications. However, the Markov chains might be easily trapped by local modes and long mixing times may be required. Moreover, MCMC methods estimate all the unknown parameters in the Bayesian model simultaneously. This is not suitable for the hierarchical multiscale Bayesian model.

In order to bridge the gap between scales and explore multi-modal posteriors efficiently, a SMC method is employed [61, 51, 6, 39, 19]. First, the idea of annealing/tempering is introduced. Given the target posterior distribution in Eq. (2.18), a sequence of auxiliary distributions, $\{\pi_0, \cdots, \pi_n\}$, is proposed to move smoothly from a tractable distribution π_0 to the target distribution $\pi_n \equiv \pi_q(\psi_q)$. We adopt the following auxiliary distributions:

$$\pi_t(\psi_q) \propto \mathcal{L}^{\gamma_t}(\psi_q|\boldsymbol{d})p(\psi_q), \tag{2.19}$$

where $t = 0, 1, \cdots, n$ and $0 = \gamma_0 < \gamma_1 < \cdots < \gamma_n = 1$ are tempering parameters. Here, $\mathcal{L}(\psi_q|\boldsymbol{d})$ is the likelihood function, $p(\psi_q)$ is the prior distribution and γ_t serves as the power exponent of the likelihood function.

The SMC method takes samples from such a sequence of probability distributions based on importance sampling and resampling techniques and constructs a sequential Bayesian inference. At step t, the basic idea is to obtain a large collection of N weighted random samples $\{\psi_{q,t}^{(i)}, \Omega_t^{(i)}\} \quad (i = 1, \cdots, N)$ (also referred to as particles) whose empirical distribution converges asymptotically to the current target distribution π_t. Each particle can be considered as a possible configuration of the system's state [45] with an importance weight.

According to Eq. (2.19), it is easy to sample directly from π_0, the prior distribution, at the initial step. The importance sampling technique is performed

sequentially to the auxiliary distributions, which is called the sequential importance sampling (SIS) [51]. In this work, we move these particles using a predetermined Markov transition kernel [61]. Suppose that at step $t-1$, we have N samples $\{\psi_{q,t-1}^{(i)}\}$ distributed in η_{t-1}. A Markov transition kernel K_t with invariant distribution π_t is proposed and new samples are marginally distributed as

$$\eta_t(\psi_q') = \int \eta_{t-1}(\psi_q) K_t(\psi_q, \psi_q') d\psi_q. \tag{2.20}$$

We use the Metropolis-Hastings kernel with invariant distribution π_t based on a random walker sampler to move the particles $\psi_{q,t-1}^{(i)}$. The $\boldsymbol{w}_q, \delta_q, \sigma_\zeta, \kappa, \phi$ in the random vector ψ_q are updated individually via a MH kernel with a normal random walk proposal.

The importance weight $\omega^{(i)}$ estimates the discrepancy between the proposal distribution $\eta_t(\psi_q)$ and the current auxiliary distribution $\pi_t(\psi_q)$. Using the MCMC transition kernel, a recursive form for the calculation of the importance weight is [61]

$$\omega_t^{(i)} = \omega_{t-1}^{(i)} \frac{\pi_t(\psi_{q,t-1}^{(i)})}{\pi_{t-1}(\psi_{q,t-1}^{(i)})}. \tag{2.21}$$

It is inevitable that the SIS algorithm will degenerate and the variance of the importance weights stochastically will increase with t [20, 52]. We measure the degeneracy using the effective sample size (ESS) calculated from the normalized importance weights [52] as:

$$\text{ESS} = (\sum_{i=1}^{N} (\Omega^{(i)})^2)^{-1}. \tag{2.22}$$

We define a threshold $\text{ESS}_{\min} = \xi N \quad (\xi < 1)$. If ESS $<$ $\text{ESS}_{\min}$, we carry out resampling to relieve the degeneracy of the algorithm. The simplest approach is the multinomial resampling which draws N new samples from $\{\psi_{q,t}^{(i)}\}_{i=1:N}$

according to the corresponding normalized weights $\{\Omega^{(i)}\}_{i=1:N}$ [61, 53]. After resampling, we obtain N weighted particles for the target posterior distribution $\pi_n(\psi_q)$. Obviously, the SMC method is directly parallelizable.

A summary of the MH kernel with a random walker sampler and of the SMC algorithm is given below in Algorithms I and II, respectively.

Algorithm I : Update of w_q: MH kernel with a random walker proposal

1. Sample $u \sim \mathcal{U}(0,1)$.
2. Sample $\widetilde{\boldsymbol{w}}_q \sim \mathcal{N}(\boldsymbol{w}_{q,t-1}^{(i)}, \sigma_m^2)$.
3. If $u < \min\{1, \frac{\pi_t(\widetilde{\boldsymbol{w}}_q)}{\pi_t(\boldsymbol{w}_{q,t-1}^{(i)})}\}$, set $\boldsymbol{w}_{q,t}^{(i)} = \widetilde{\boldsymbol{w}}_q$.
4. Else set $\boldsymbol{w}_{q,t}^{(i)} = \boldsymbol{w}_{q,t-1}^{(i)}$.

Algorithm II : SMC algorithm

1. Initialization: For $i = 1, \cdots, N$, sample $X_0^{(i)} \sim \pi_0(x)$ and set the importance weight $\omega_0(\psi_{q,0}^{(i)}) = \frac{1}{N}$.
2. Updating: At time t, for $i = 1, \cdots, N$, sample $\psi_{q,t}^{(i)} \sim K_t(\psi_{q,t-1}^{(i)}, \cdot)$ and set the importance weight according to Eq. (2.21). Then normalize the importance weight by $\Omega_t^{(i)} = \frac{\omega_t^{(i)}}{\sum_{k=1}^N \omega_t^{(k)}}$.

3. Resampling: Calculate the effective sample size (ESS) by Eq. (2.22). If ESS $<$ $\text{ESS}_{\min}$, resample the particles $\{\psi_{q,1:n}^{(i)}, \Omega_n^{(i)}\}$ according to $\{\Omega_n^{(i)}\}$ to obtain a new population $\{\psi_{q,1:n}^{(i)}, 1/N\}$.

4. Repeat the above steps until $t = n$, i.e. the particles are distributed in the last distribution in the sequence.

2.3 Hierarchical Bayesian model

In the earlier sections, we defined the standard sparse grid on various levels of resolution. Given a sparse grid, the spatially varying parameter can be represented by the basis functions associated with the collocation points. Then the unknown surpluses are treated as random variables and a Bayesian model can be constructed to make inference from the observation data. Based on the one-scale Bayesian model on a single grid, we propose a hierarchical, multi-scale Bayesian model in this section. A set of sparse grids from coarse to fine scales are constructed through adaptive refinement. On each grid, a one-scale Bayesian model is defined. Due to the hierarchical structure of the collocation points, a sequential estimation of the surpluses can be performed from coarse to fine scales. Only part of the surpluses need to be estimated on a certain grid, which significantly reduces the dimensionality of the inverse problem. Also, the surpluses can serve as an indicator of the smoothness of the parameter field. More support nodes will be added on non-smooth regions and rapid changes in the parameter field can be effectively captured. In this way, an optimal choice of the basis functions can be achieved.

2.3.1 Adaptive sparse grid

For the Newton-Cotes sparse grid at level q, the set of points can be obtained by refining the grid of level $q-1$ in a principled way. In fact, the $1D$ equidistant points of the sparse grid can be considered as a tree-like data structure as shown in Fig. 2.2. We can consider the interpolation level of a grid point m as the depth

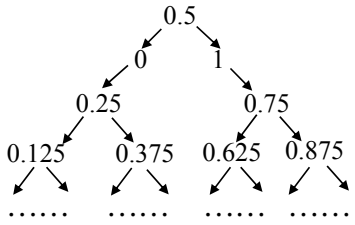

Figure 2.2: One-dimensional tree-like structure of the sparse grid.

of the tree. We denote the father of a grid point as $F(m)$, where the father of the root 0.5 is itself, i.e., $F(0.5) = 0.5$. There are two sons for each grid point in each dimension. For a grid point in an N-dimensional space (here $N = 1, 2, 3$ for spatial fields), there are $2N$ sons. The sons are also the *neighbor points* of the father. The neighbor points are just the support nodes of the hierarchical basis functions in the next interpolation level. By adding the neighbor points, we actually add the support nodes from the next interpolation level. Therefore, we refine the grid locally while not violating the developments of the Smolyak algorithm [56].

In this way, a set of intermediate grids can be obtained between two standard levels of sparse grid. Fig. 2.3 presents a 2D example in which the standard sparse grid at level 2 is constructed from the grid at level 1 by pointwise refinement. We arbitrarily pick up a point at level 1 and add its $2N$ neighbor points to the sparse grid to obtain a finer, intermediate grid. This procedure contin-

ues until we go through all collocation points that belong to level 1 sparse grid. Since it is possible that the neighbors of one point have already been generated by other points, the difference in the number of collocation points of two successive intermediate grids is no more than $2N$, e.g. 4 in a 2D case. Each intermediate grid provides a set of basis functions for parameterizing the spatially varying parameter. Rather than working on the standard sparse grid, we define Bayesian models on intermediate grids and make sequential inference about the surpluses. More details will be given in the next section.

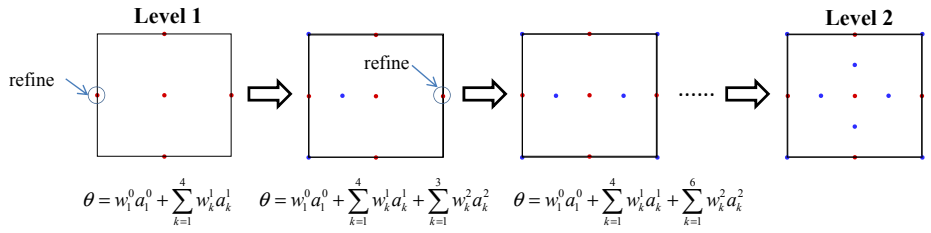

Figure 2.3: An example of refining the sparse grid in a two-dimensional domain.

Furthermore, an adaptive refinement strategy can be used to reduce the collocation points on the sparse grid and thus the dimensionality of the inverse problem. In fact, the hierarchical surplus is a natural candidate for detecting non-smooth regions [54]. Here, the basic idea for adaptivity is to use the posterior mean of hierarchical surpluses as an error indicator to detect the smoothness of the spatially varying parameter field estimated on the current grid. We only refine the hierarchical basis functions a_j^i whose magnitude of the surplus satisfies $|w_{\mathbf{j}}^{\mathbf{i}}| \geq \varepsilon$. If the criterion is satisfied, the $2N$ neighbor points are added into the current sparse grid. Otherwise, we assume that the local region is smooth enough and further refinement on this basis is not required. With $\varepsilon > 0$ the parameter that controls the adaptive refinement, we introduce the following Algorithm III:

Algorithm III : Adaptive sparse grid representation of the unknown parameter

repeat

1. Set the initial level of sparse grid as q and construct a standard sparse grid. Construct a full Bayesian model on this level and infer all the surpluses from observation data (Section 2.2).

2. Calculate the posterior mean of the surpluses.

3. Put the collocation points with surpluses $|w_{\mathbf{j}}^{\mathbf{i}}| > \varepsilon$ in the active node set which denotes the set of points whose 'sons' defined in Eq. (**??**) would be added to refine the original grid.

while the active node set is not empty **do**

(a) Pick up a point and add its $2N$ neighbor points to the sparse grid. An intermediate grid is obtained.

(b) Construct a full Bayesian model and infer all the surpluses at the current grid.

(c) Remove the point from the active node set.

end while

4. Set $q = q + 1$.

5. Place all newly added collocation points into the empty active node set

until $q = q_{\max}$ or all the points in the active node set have surpluses $|w_{\mathbf{j}}^{\mathbf{i}}| < \varepsilon$, i.e. the refinement of the grid terminates.

2.3.2 Hierarchical Bayesian inference

The hierarchically structured sparse grid provides a multiscale representation of the spatially varying parameter (Fig. 2.4). Once a grid is defined, we can

construct a Bayesian model for the full surpluses and estimate them from observation data. In Section 2.2.2, the SMC method was used to explore the posterior distribution on a single grid. In this section, we propose a strategy that bridges the estimation of surpluses on different scales. As a result, the dimensionality of the inverse problem is further reduced and informative priors are developed on a fine grid by incorporating the information from the posterior at the coarser grid.

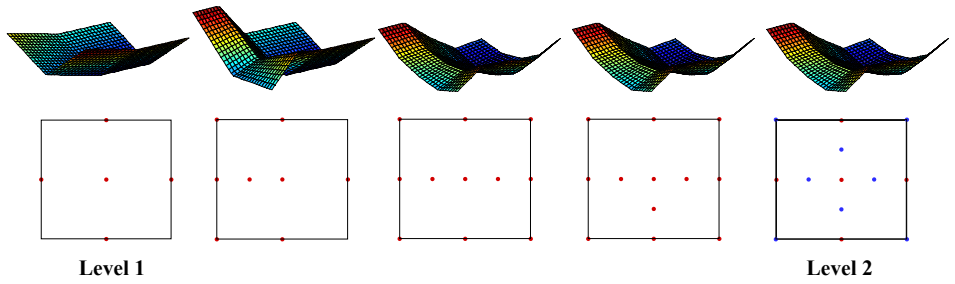

Figure 2.4: A hierarchical representation of the spatially varying parameter on different scales.

Consider a sparse grid indexed by (q, l) where l denotes the l-th intermediate grid between level q and $q+1$. Due to the nested fashion of the grid, the collocation points are composed of two parts: the inherited points from the immediate coarser grid and the new points obtained by refinement. Thus the full surpluses can be written as

$$\boldsymbol{w}_{q,l} = \{\boldsymbol{w}_{q,l-1}, \boldsymbol{w}^{\star}_{q,l}\}, \tag{2.23}$$

where $\boldsymbol{w}_{q,l-1}$ is a vector of surpluses which have been estimated on the coarse grid, and $\boldsymbol{w}^{\star}_{q,l}$ is a vector of new surpluses in the current grid. In the one-scale Bayesian model in Section 2.2, a non-informative prior is assumed for the surpluses. However, the posterior estimation of $\boldsymbol{w}_{q,l-1}$ on the coarse grid $(q, l-1)$ provides plenty of information for the re-estimation of the surpluses on grid

(q, l). In fact, in a direct sparse grid interpolation at level q, we keep the surpluses estimated from levels 0 to $q-1$ and only estimate the surpluses of new collocation points at level q [54, 42]. Now the prior for $\boldsymbol{w}_{q,l}$ is defined as

$$p(\boldsymbol{w}_{q,l}) = p(\boldsymbol{w}_{q,l-1})p(\boldsymbol{w}^{\star}_{q,l}) \propto p(\boldsymbol{w}_{q,l-1}|\boldsymbol{d})p(\boldsymbol{w}^{\star}_{q,l}), \tag{2.24}$$

where a multivariate Gaussian distribution is assumed for the prior $p(\boldsymbol{w}^{\star}_{q,l})$, i.e. $p(\boldsymbol{w}^{\star}_{q,l}) \sim \mathcal{N}(0, \sigma^2_{w,ql}\mathbf{I})$. The posterior distribution $p(\boldsymbol{w}_{q,l-1}|\boldsymbol{d})$ at the coarse grid $(q, l-1)$ is taken as the prior. The samples from the posterior are directly used as the samples from the prior $p(\boldsymbol{w}_{q,l-1})$ at the current grid. This hierarchical inference strategy is depicted in Fig. 2.5.

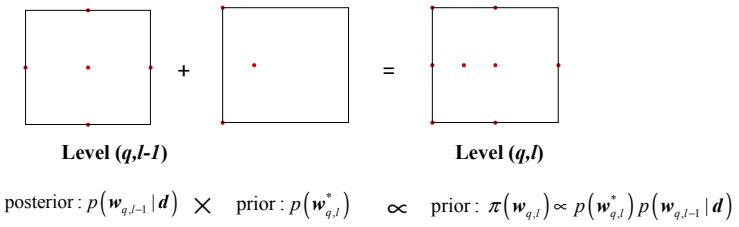

Figure 2.5: A hierarchical inference by taking the posterior $p(\boldsymbol{w}_{q,l-1}|\boldsymbol{d})$ as the prior on a refined grid.

For the surpluses from points of the previous levels of interpolation, $\boldsymbol{w}_{q,l-1}$, not all elements need to be re-estimated on grid (q, l). When a surplus satisfies $|w^i_j| < \varepsilon$, no re-estimation is required since the corresponding basis function makes negligible contribution to the interpolation. For other surpluses, if the posterior mean after re-estimation is close to that before re-estimation, no further re-estimation is required. In this study, we use the same ε as a criterion to determine convergence. After we finish estimating the surpluses on the current grid, the adaptive refinement strategy introduced in Algorithm III is carried out to refine the grid. Then the hierarchical Bayesian inference is performed on the finer grid.

2.4 Numerical examples

2.4.1 Problem definition

We consider the nonlinear inverse problem of estimating the permeability field in flow in porous media. First, a physical model is built for corner-to-corner flow in a 2D unit square domain $D = [0, 1]^2$ [69, 56]. The injection and production wells are located in diagonally opposite vertices of the grid (Fig. 2.6).

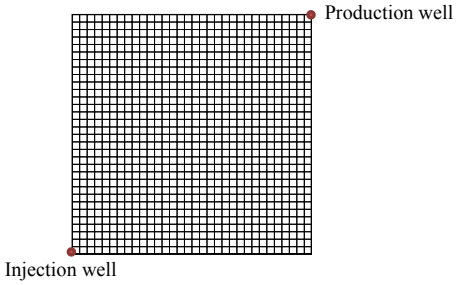

Figure 2.6: Schematic of the quarter five-spot problem.

The governing equations for the flow velocity and pressure are:

$$\nabla \cdot \boldsymbol{u}(\boldsymbol{x}) = f(\boldsymbol{x}), \tag{2.25}$$

$$\boldsymbol{u}(\boldsymbol{x}) = -k(\boldsymbol{x})\nabla p(\boldsymbol{x}), \tag{2.26}$$

where $f(\boldsymbol{x})$ here is the source/sink term, $\boldsymbol{u}$ is the velocity given by Darcy's law and p is the pressure. An isotropic permeability field is assumed and denoted by $k(\boldsymbol{x})$. All boundaries are no-flow boundaries.

In the inverse problem of interest, the permeability field is the unknown parameter to be inferred from flow or pressure data at finite number of sensor locations. A mixed finite element method is used to obtain the numerical solu-

tion on a 32×32 grid. The observation data are generated from the numerical solutions by adding simulated noises.

To keep the permeability non-negative, we will treat the logarithm of the permeability, $\log(k)$, as the main unknown of the inverse problem. $N = 1200$ particles are employed in the implementation of the SMC algorithm. The threshold of ESS is set to be $\text{ESS}_{\min} = 0.85N$. A linear cooling schedule is selected for γ_t in Eq. (2.19). For 1500 time steps, the sequence $\{\gamma_0, \cdots, \gamma_{1500}\}$ increases uniformly from 0 to 1. We take the maximum level of sparse grid $q_{\max} = 5$ and the parameter for the adaptive refinement $\varepsilon = 0.05$ (in Section 2.3.1). The hyperparameters used in the prior and likelihood are: $\alpha_0 = 0.1, \beta_0 = 10$, $\alpha_1 = 0.01, \beta_1 = 100$, $\alpha_{\sigma_\zeta} = 0.1, \beta_{\sigma_\zeta} = 10$, $\alpha_\kappa = 0.01, \beta_\kappa = 100$, $\alpha_\phi = 0.01, \beta_\phi = 100$. The initial values for the unknown surpluses are generated from the priors, while the initial values for the mean of the model error vary with the level of sparse grid.

Example 1

In the first example, we consider a simple permeability field of the following form [49, 56]:

$$\log k(x, y) = 2(x - 0.5) + 2(y - 0.5). \tag{2.27}$$

We apply a one-scale Bayesian inference to estimate the permeability. The objective of this test example is to examine the efficiency of sparse grid representation as well as the SMC algorithm proposed. For the representation of smooth functions, the sparse grid method is superior since it requires a small number of basis functions. In this example, the sparse grid of level 1 with only 5 collocation points is capable of representing the log-permeability field exactly. Thus we make the inference on this sparse grid and model error is not taken into ac-

count. The pressure is measured at a 5×5 evenly distributed sensor network and examine two cases with 2% and 5% relative noise level.

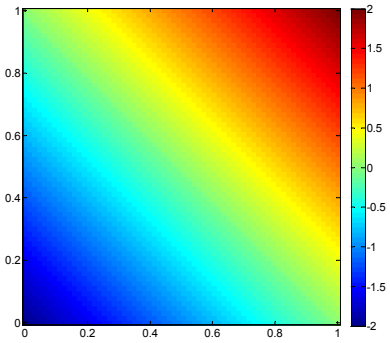

Figure 2.7: True permeability (logarithm) in Example 1.

Posterior quantiles of the log-permeability inferred from the two datasets are plotted in Figs. 2.8 and 2.9. It is seen that the basic Bayesian framework based on sparse grid and SMC provides rather good estimates of such a smooth permeability field. When the level of measurement noise is reduced, the inferred estimates are improved. The same problem was studied using MCMC in [56] where the permeability field was approximated by Gaussian kernels. 25 kernels were used to provide reasonable estimates. To make sure the Markov chain converges, 50000 iterations were carried out. In this work, much fewer basis functions are required for a good representation of the permeability field. Besides, the particles can run in parallel, each with 1500 MCMC updates. Clearly, this approach largely reduces the computation cost.

Example 2

In this example, the logarithm of the true permeability (Fig. 2.10) is generated from a Gaussian process with a correlation function $\rho(r) = \exp(-r^2)$ where r is

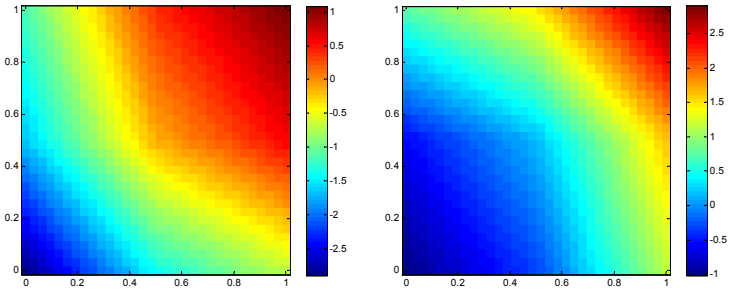

Figure 2.8: Posterior quantiles of the log-permeability (2% noise): 5% quantile (left) and 95% quantile (right).

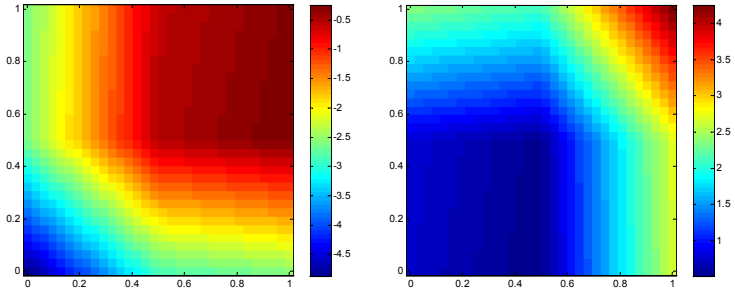

Figure 2.9: Posterior quantiles of the log-permeability (5% noise): 5% quantile (left) and 95% quantile (right).

the distance between two locations. The pressure is measured at a 5×5 sensor network with 2% noise. The initial values for the mean of the model error, δ_q, are set to be $\frac{2}{2^l}\%$ of the mean of observation data, where l is the level of sparse grid interpolation.

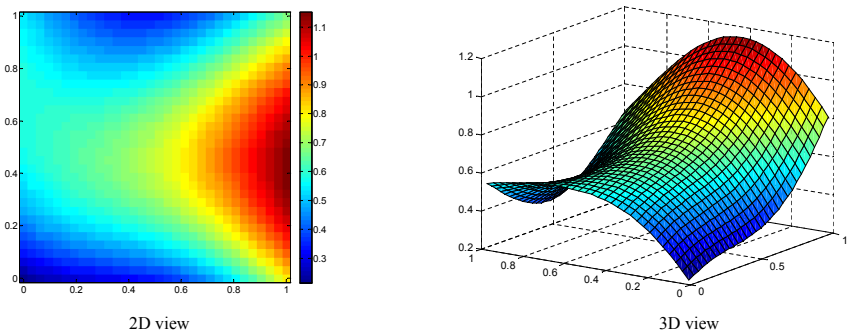

Figure 2.10: True permeability (logarithm) generated from a Gaussian Process.

Two models for model error δ as discussed in Section 2.2.1 are applied here. The posterior means computed for different levels of the sparse grid are depicted in Fig. 2.11 and Fig. 2.12. For such a smooth permeability field, sparse grid of level 2 is enough to provide a good approximation and the Bayesian inference is close to the true permeability field. In Table 2.1, the posterior means of the estimated model errors $\delta_q^{(1)}$ and $\delta_q^{(2)}$ corresponding to the two models are compared with the true values $\delta_q^\star$ which is taken as the mean of the difference between the observation data and those predicted by the approximate model $\boldsymbol{F}_q$. We can see that both models give similar results except that Model 1 gives better inference at the upper left corner of the permeability field. This can be easily understood because the dependence among model errors at different locations are considered in Model 1. However, the evaluation of hyperparameters κ and ϕ in Fig. 2.13 shows that the spatial variance is restricted in a relatively small range and scale, which implies small $\kappa H(\phi)$ in Eq.(2.13). While intuitively the model errors are correlated, the adaptivity of sparse grid in the hierarchical Bayesian model weakens the correlation. Suppose a location where the model error is large, more collocation points would be added around it. As a result, the model error is somehow localized and there is only weak correlation between model errors at different locations. In Fig. 2.14, the posterior quantiles on level 3 obtained from the simplified Model 2 are presented. This shows that even when model errors are uncorrelated, we can still obtain reasonable inference from observation data.

Table 2.1: Posterior mean of the model error $\delta_q^{(1)}, \delta_q^{(2)}$ and true values $\delta_q^\star$

	Level 1	Level 2	Level 3
$\delta_q^{(1)}$	0.187	0.0281	0.0099
$\delta_q^{(2)}$	0.113	0.0205	0.0116
$\delta_q^\star$	0.0821	0.0184	0.00846

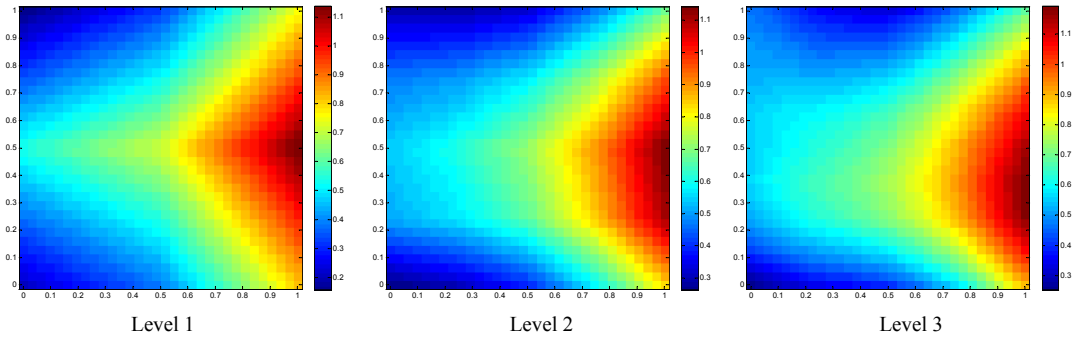

Figure 2.11: Posterior means estimated on three levels of the sparse grid (Model 1).

Example 3

In this example, the log-permeability field is generated based on a variogram model using the software *snesim* [71]. The field is defined on a 32×32 grid with a constant value of permeability in each element. The true log-permeability is plotted in Fig. 2.15. The pressure is measured on a 5×5 sensor network with 2% noise. The initial values for the mean of the model error, δ_q, are set as in Example 2.

The multiscale Bayesian inference is performed on four levels of the sparse grid. The posterior means with respect to the two models (model error) are depicted in Figs. 2.16 and 2.17. The estimated model errors are listed in Table 2.2.

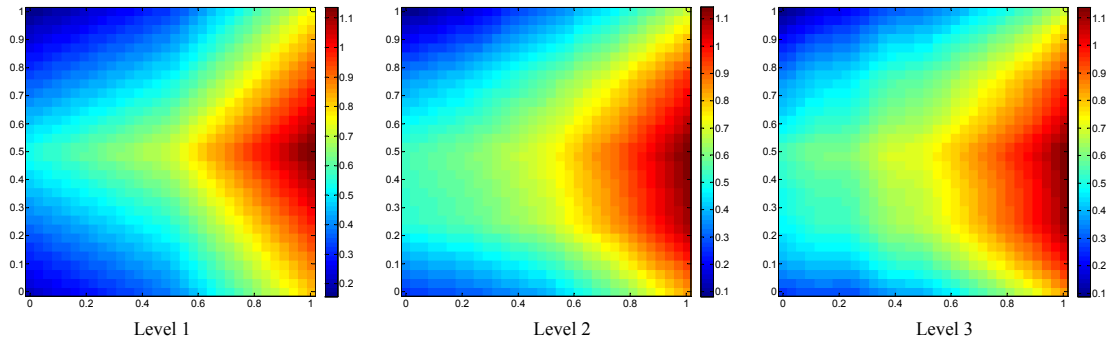

Figure 2.12: Posterior means estimated on three levels of the sparse grid (Model 2).

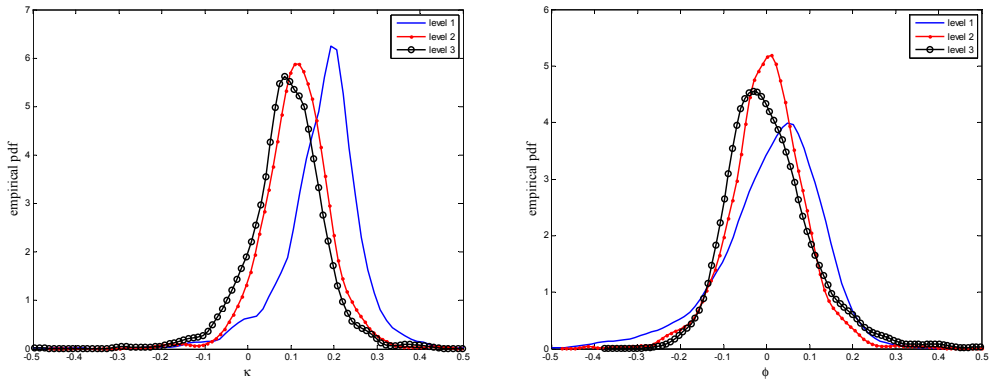

Figure 2.13: Empirical pdf of κ (left) and ϕ (right) at different levels of the sparse grid.

It is seen that even on a coarse scale (level 2), the main features of the true permeability are efficiently captured by the sparse grid and correctly inferred from the limited observation data. The empirical pdf of hyperparameters κ and ϕ in Model 1 are given in Fig. 2.18. As in Example 2, the two parameters indicate a weak correlation among model errors at different locations. The posterior quantiles on level 4 obtained from Model 2 are shown in Fig. 2.19.

In Examples 2 and 3, good estimates can be obtained at a relatively coarse level of resolution. However, due to the high nonlinearity of the forward model,

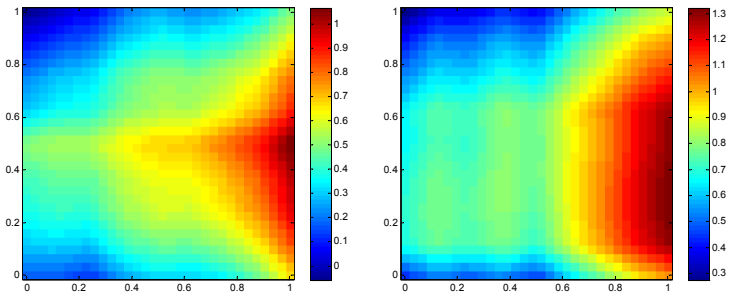

Figure 2.14: Posterior quantiles of the log-permeability (Model 2): 5% quantile (left) and 95% quantile (right).

Table 2.2: Posterior mean of the model error $\delta_q^{(1)}, \delta_q^{(2)}$ and true values $\delta_q^\star$

	Level 1	Level 2	Level 3	Level 4
$\delta_q^{(1)}$	-0.870	-0.0931	-0.0154	-0.0073
$\delta_q^{(2)}$	-0.621	-0.0710	-0.0215	-0.0103
$\delta_q^\star$	-0.734	-0.0807	-0.0174	-0.00862

the forward problem is multi-modal and the standard MCMC performs poorly in these cases. For demonstration, the Metropolis-Hastings algorithm is directly applied for inference from accurate data (no noise included) on sparse grid at level 3. The initial values of surpluses are set to be zero. In Fig. 2.20, the posterior mean after $50,000$ iterations is plotted. We can see that the Markov chain is trapped by a local mode.

Example 4

In this example, we apply the multiscale Bayesian inference to a channelized permeability field. Channelized permeability is generally difficult to resolve by conventional models, such as the GP model and the K-L (Karhunen-Loève)

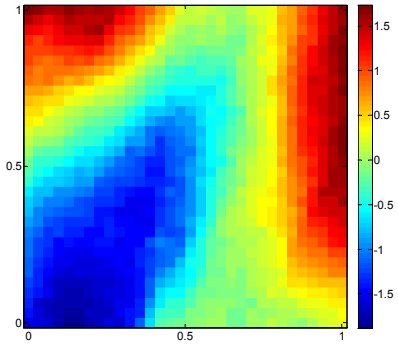

Figure 2.15: True permeability (logarithm) generated using the software *snesim*.

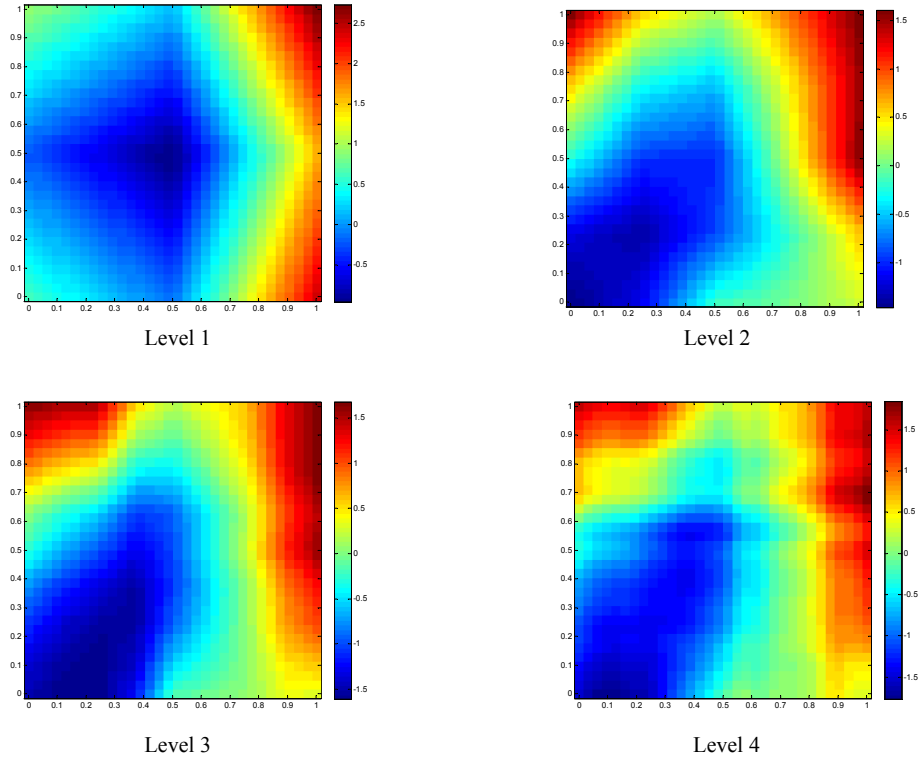

Figure 2.16: Posterior means estimated on four levels of the sparse grid (Model 1).

expansion. Fig. 2.21 gives four realizations of the channelized permeability created by the software *snesim*. The log-permeability values are at 1 in black regions and 0 in white regions. In this example, the permeability plotted in Fig. 2.21(b) is set to be the true permeability. The pressure is measured on a 20×20 evenly

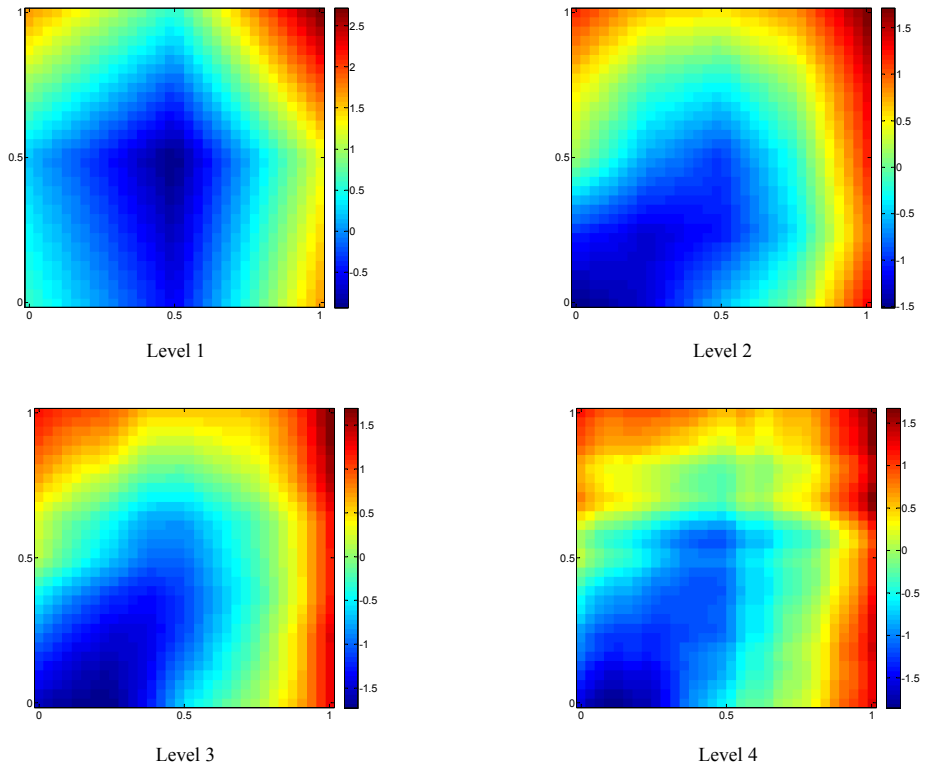

Figure 2.17: Posterior means estimated on four levels of the sparse grid (Model 2).

distributed sensor network with 2% and 5% noise. The initial values for the mean of the model error, δ_q, are set to be $\frac{5}{2^l}\%$ of the mean of observation data. All the permeability fields in this example are defined on 32×32 gridblocks. According to the weak correlation among model errors shown in Examples 2 and 3, we only consider the simplified Model 2 in this example.

At first, we make Bayesian inference on the standard sparse grid, i.e. no adaptive strategies are used to refine the grid. The posterior means estimated from coarse to fine scales are presented in Fig. 2.22. The number of collocation points of the five levels are 5, 13, 29, 65 and 145. With the increase of collocation points, more local features of the true permeability field are captured in the estimation. The black strip on the bottom is well captured after level 3. However,

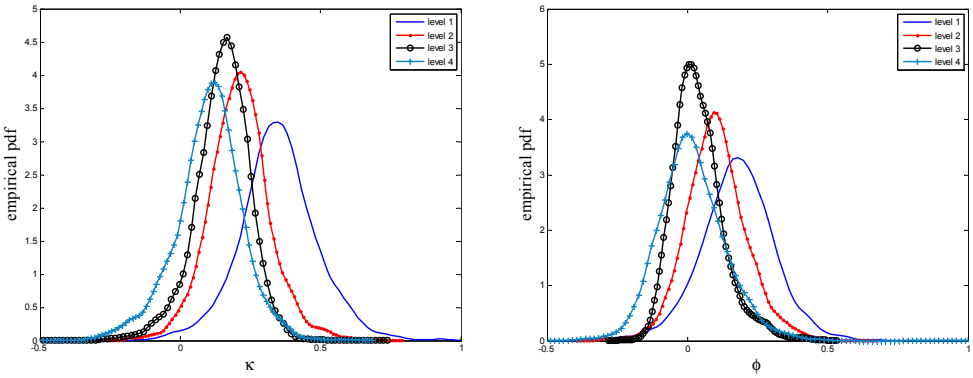

Figure 2.18: Empirical pdf of κ (left) and ϕ (right) at different levels of the sparse grid.

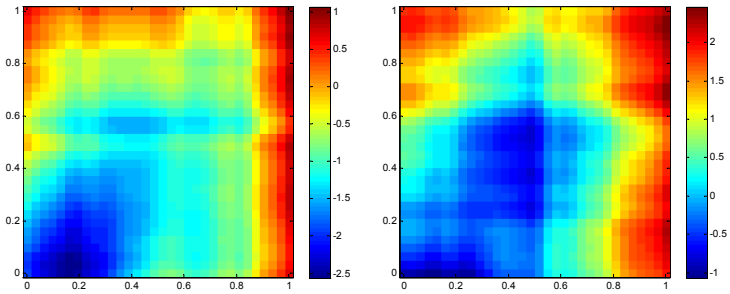

Figure 2.19: Posterior quantiles of the log-permeability: 5% quantile (left) and 95% quantile (right).

although there is some trend for the black strip on the left top corner on level 2, it soon disappears on subsequent finer grids.

We next apply the adaptive strategy in the refinement of the sparse grid. In this case, we start the inference on a standard sparse grid of level 2. The collocation points with (posterior mean) surpluses whose magnitude is less than 0.05 are removed from the grid and thus rejected from further refinement. As a result, the dimensionality of the inverse problem is largely reduced. The estimated posterior means and corresponding adaptive sparse grids are depicted in Fig. 2.23. The posterior quantiles on level 5 are shown in Fig. 2.24. The estimated model errors are listed in Table 2.3. We can see that the results are

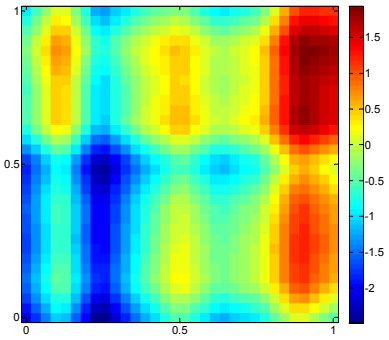

Figure 2.20: Posterior mean of log-permeability estimation by MCMC.

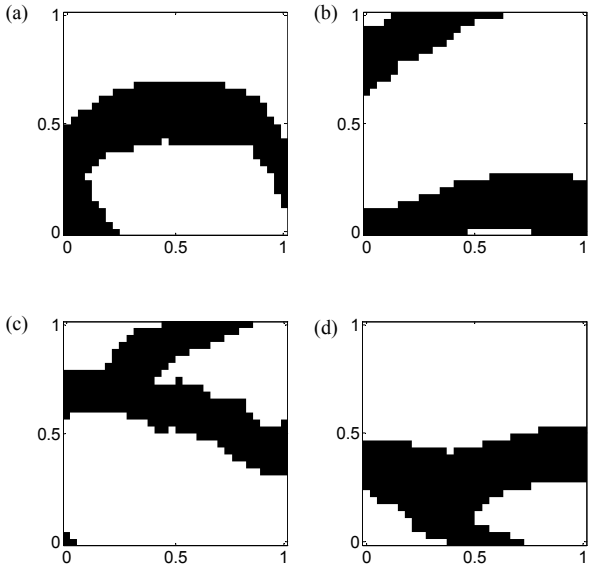

Figure 2.21: Examples of channelized permeabilities. The log-permeability values are 1 in black regions and 0 in the white regions.

improved after reducing the number of collocation points. The two black strips in the true permeability field are captured. The unsatisfactory results from the standard sparse grid representation may be due to overfitting. In this example, the true permeability field is composed of several smooth segmentations. The discontinuity only occurs on the sharp edges. Although the Smolyak algorithm has made general optimization in the choice of collocations points, there are still

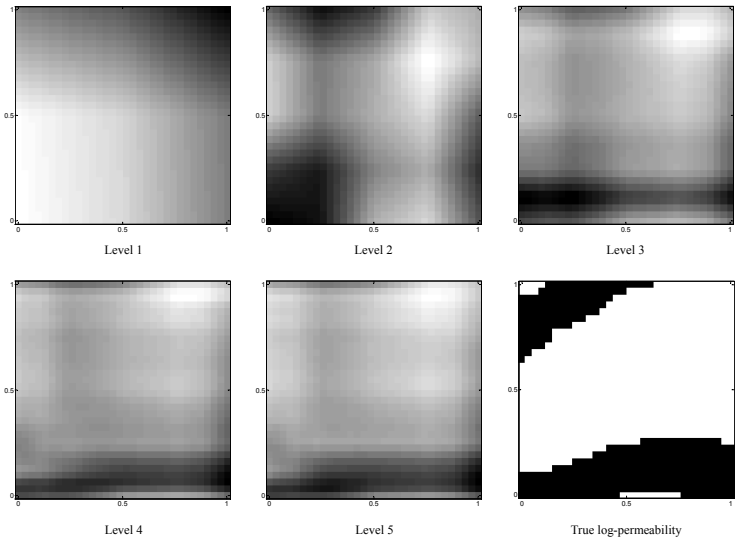

Figure 2.22: Posterior means estimated on the standard sparse grid from level 1 to level 5 (2% noise in data).

redundant points for this specific problem. This may produce erroneous values on the unnecessary points and thus affect the overall configuration of the estimated parameter field. The same strategy is also applied to make inference from data with 5% noise (Figs. 2.25 and 2.26). It should come as no surprise that the estimates are not as good as those inferred from data with lower level of noise.

Table 2.3: Posterior mean of the model error δ_q and true values $\delta_q^\star$ (2% noise)

	Level 2	Level 3	Level 4	Level 5
δ_q	-0.189	-0.155	-0.0632	-0.0118
$\delta_q^\star$	-0.139	-0.132	-0.0514	-0.0133

Table 2.4: Posterior mean of the model error δ_q and true values $\delta_q^\star$ (5% noise)

	Level 2	Level 3	Level 4	Level 5
δ_q	-0.292	-0.279	-0.156	-0.0407
$\delta_q^\star$	-0.312	-0.253	-0.131	-0.0350

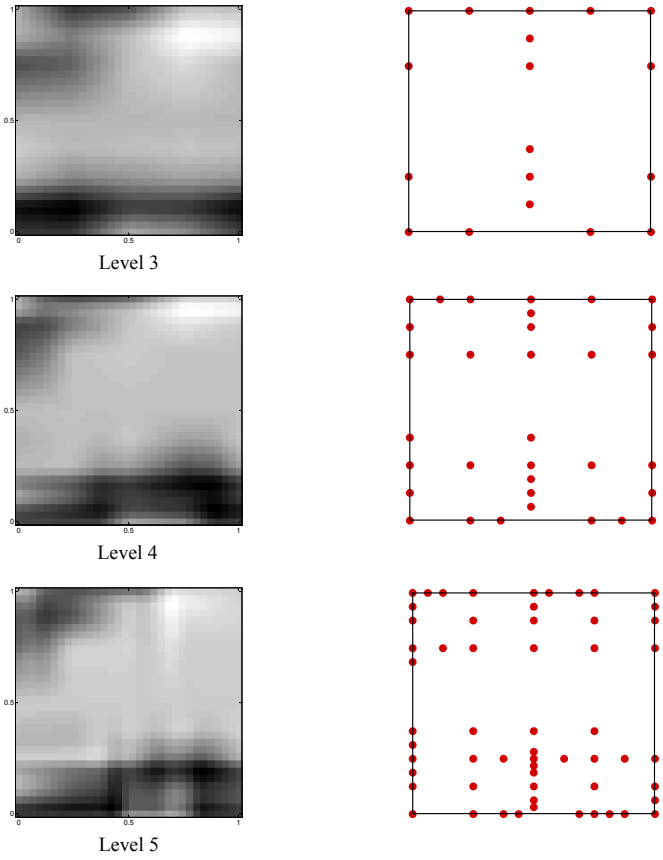

Figure 2.23: Posterior means estimated on sparse grid with adaptive refinement (2% noise in data).

2.5 Conclusions

A multiscale Bayesian framework for the identification of spatially varying parameters was introduced. The parameter field was discretized using a sparse

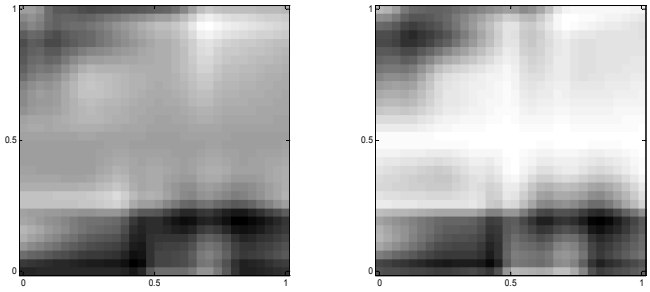

Figure 2.24: Posterior quantiles of the log-permeability (2% noise): 5% quantile (left) and 95% quantile (right).

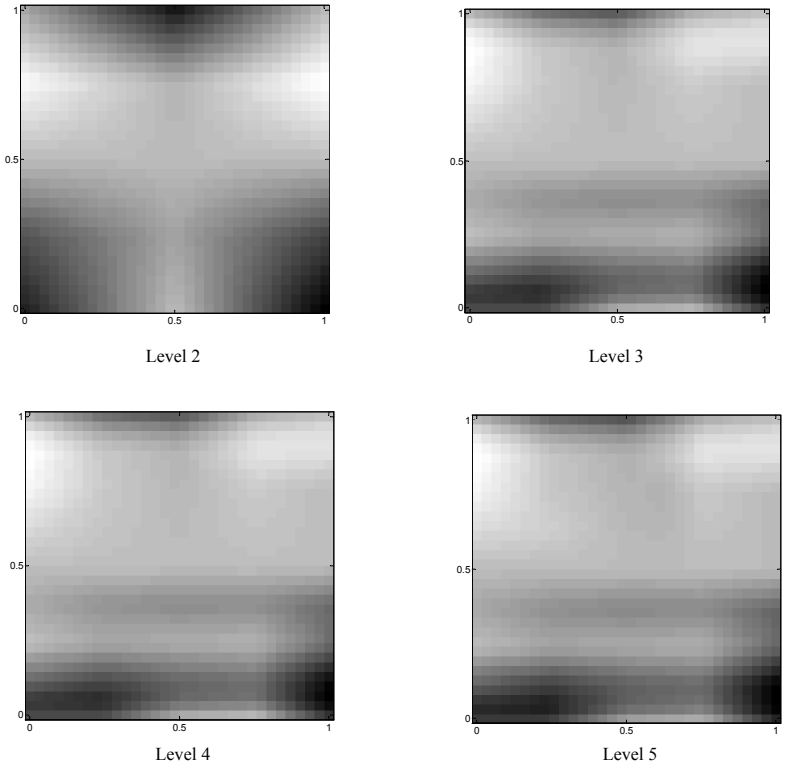

Figure 2.25: Posterior means estimated on sparse grid with adaptive refinement (5% noise in data).

grid and represented by local basis functions associated with the collocation points. Based on the hierarchical property of the sparse grid, a multiscale representation of the parameter field was introduced and a sequence of hierarchical Bayesian models from coarse to fine scales.

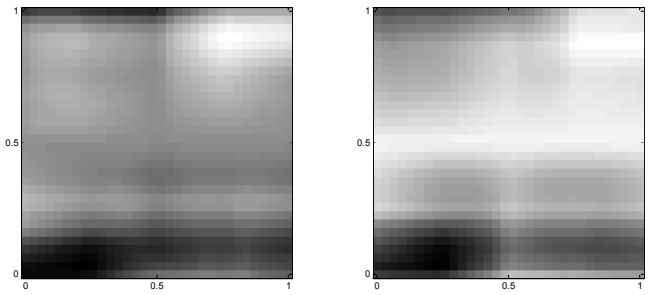

Figure 2.26: Posterior quantiles of the log-permeability (5% noise): 5% quantile (left) and 95% quantile (right)

The sparse grid provides an effective way of finding an optimal choice of basis functions to approximate the spatially varying parameter. The estimated coefficients (surpluses) of the basis functions serve as indicators of the interpolation error. Accordingly, the refinement of the grid can be performed in an adaptive way. This reduces the dimensionality of the inverse problem and the computational cost of Bayesian inference. The adaptive refinement strategy removes unimportant points from estimation, thus avoids possible overfitting and leads to improved results. The multiscale Bayesian framework proposed is a generalized way of estimating unknown spatially varying parameters from observation data. Without any prior information, the sparse grid method provides an effective strategy to construct an optimal Bayesian model.

The SMC algorithm used here is directly parallelizable and well suited for multi-modal problems. Using standard MCMC, the estimation of the surpluses can be trapped by many local modes at coarse levels of resolution, which could result in failure of further inference on fine scales. The samples from the posterior distributions at a coarse grid can provide prior information for inference of the surpluses of these collocation points on a finer grid, which helps to improve the quality of the calculations and to speed up the convergence rate.

CHAPTER 3

CONSTRUCTING HIGH-DIMENSIONAL STOCHASTIC INPUT MODEL WITH PROBABILISTIC GRAPHICAL MODELS

In most forward problems of multiscale systems, the stochastic input random field is in a high-dimensional space and it is desirable to map the input random field to a lower-dimensional space to improve the efficiency of uncertainty quantification. On the other hand, we need to make sure that the reconstruction of input random field from a lower-dimensional representation is as accurate as possible. For this purpose, the KL expansion has been widely used in stochastic reduced-order modeling. For non-Gaussian random fields, the modeling of KL expansion coefficients presents a number of computational challenges. Due to the curse of dimensionality, the underlying dependence relationships between these coefficients are difficult to capture. As a result, we propose a graphical model based approach to learn the dependence by running a number of conditional independence tests using observation data. Thus a probabilistic model of the joint probability density function (PDF) is obtained and it is factorized into a set of conditional distributions based on the dependence structure of the variables. The estimation of the joint PDF from data is then transformed to estimating conditional distributions under reduced dimensions. To improve the computational efficiency, a polynomial chaos expansion is further applied to represent the random field in terms of a set of standard random variables. This chapter closely follows the work in [90].

3.1 Problem definition

Let us consider a complete probability space $(\Omega, \mathcal{F}, \mathcal{P})$ where Ω is the sample space, $\mathcal{F}$ the σ-algebra of subsets in Ω and $\mathcal{P}$ the probability measure. A random field on a bounded spatial domain D is denoted by $\boldsymbol{a}(\boldsymbol{x}, \omega)$. Generally, the random field is associated with a discretization of the spatial domain and can be represented by a random vector, $\boldsymbol{a} \doteq (a_1, \cdots, a_p)^T : \Omega \to \mathbb{R}^p$. Each random variable a_i represents the property of a grid block in the discretized domain. Since the dimensionality of the random field could be very high in many cases, it is desirable to find a reduced-order representation $\boldsymbol{\eta} \in \mathbb{R}^q$ such that $q < p$. By drawing samples of $\boldsymbol{\eta}$ in a lower-dimensional space, we obtain realizations of underlying random field $\boldsymbol{a}$.

Various model reduction techniques have been proposed in past decades for this purpose. Many of them require estimating the joint distribution of random variables from observation data. A typical example is the construction of a reduced-order polynomial chaos representation based on KL expansion. It is necessary to get the joint distribution of coefficients in KL expansion. However, this might be challenging due to the curse of dimensionality as well as limited number of observation data. Therefore, we introduce a graphical model representation of joint distributions, which factorizes the target distribution into lower-dimensional conditional distributions. Then this method is combined with conventional model reduction techniques to given a stochastic input model in uncertainty quantification.

3.2 Probabilistic model of multivariate distributions

In this section, we construct a probabilistic model of joint distribution of random vector $\boldsymbol{\eta}$ based on a special graphical model — Bayesian network. As will discussed in section 3.2.1, a Bayesian network explicitly represents dependencies among random variables and allows us to concisely represent a full joint probability distribution. The random variables η_i are treated as nodes in the graphical model. Given samples of random vector $\boldsymbol{\eta}$, the objective is to infer the dependence structure which is represented by directed edges that link pairs of nodes. Thus the joint distribution is characterized by the structure of graphical model. In section 3.2.2, an efficient Bayesian network structure learning algorithm is introduced. The foundation of this algorithm is testing conditional independence (CI) relationships among random variables. As the test is performed locally, only a small number of observation data is required.

3.2.1 Brief introduction to Bayesian network and conditional independence

A Bayesian network, $G = \{V, E\}$, is a directed acyclic graphical model that encodes the joint probability of a set of random variables (continuous or discrete or the mixture). It combines both probability theory and graph theory for multivariate statistical modeling [23, 86, 44]. The graphical model comprises of nodes V and directed edges E that link the nodes. Each node, $v \in V$, represents a random variable, and the edges express probabilistic relationships between these variables. The parents of a variables v, denoted by Π_v, are the set of variables

that are the source of directed edges pointing to v. The neighbors or adjacent variables of v, denoted by *Adjacent*(v), are those connected with v regardless of the direction of edges. The main advantage of using Bayesian network here is its ability to factorize the joint distribution over all of the random variables into a product of factors each depending only on a subset of the variables according to the structure of the underlying graph. Take continuous random variables for instance, the joint probability of $\boldsymbol{\eta}$ can be factorized as [8, 30]

$$p(\boldsymbol{\eta}) = \prod_{i=1}^{m} p(\eta_i | \Pi_{\eta_i}) \tag{3.1}$$

The structure of a Bayesian network represents the conditional independencies among random variables (nodes). Consider three random vectors (each corresponding to a set of nodes that is nonintersecting with another), $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$. $\boldsymbol{Y}$ is independent of $\boldsymbol{Z}$ given $\boldsymbol{X}$ If

$$p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{Y}|\boldsymbol{X})$$

This relationship is formulated as $\boldsymbol{Y} \perp \boldsymbol{Z}|\boldsymbol{X}$. The conditional independence relations are associated with certain graphical structures in a Bayesian network, which is summarized by the *d*-separation criterion [8]. Consider a path between two nodes. It is said to be blocked by a set of nodes $\boldsymbol{X}$ if and only if (1) the path contains a chain that is either head-to-tail ($y \to x \to z$) or tail-to-tail ($y \leftarrow x \to z$) at the node $x \in \boldsymbol{X}$, or (2) the path contains a chain that is head-to-head ($y \to x \leftarrow z$) such that neither the middle node x, nor any of its descendants, is in $\boldsymbol{X}$. $\boldsymbol{X}$ is said to *d*-separate $\boldsymbol{Y}$ from $\boldsymbol{Z}$ if and only if $\boldsymbol{X}$ blocks every path from a node in $\boldsymbol{Y}$ to a node in $\boldsymbol{Z}$. The corresponding joint distribution satisfies the conditional independence $\boldsymbol{Y} \perp \boldsymbol{Z}|\boldsymbol{X}$.

3.2.2 Bayesian network structure learning

To estimate the probabilistic model of random variables, a BN is formed such that each variable is represented by a node. Given i.i.d. realizations of $\boldsymbol{\eta}$, the objective is to identify a directed graph G which represents the factorization of the joint PDF in Eq (3.40). Generally, there are two categories of BN structure learning algorithms. One is *search-and-score* approach, which assigns a score to each possible BN structure and find one that maximizes the score given observation data [31]. The other one is called *constraint-based* algorithm, which learns the BN structure by running local conditional independence tests to identify a dependency model containing independence relationships among random variables [14]. The learned independence assertions are taken as constraints to the final BN structure. The constraint-based algorithm selects a structure that is consistent with those constraints. In general, *search-and-score* algorithms find high-likelihood structures but do not enforce conditional independence relationships. The number of possible structures increases exponentially with the number of nodes. On the other hand, by constraining BN structure with conditional independence, *constraint-based* algorithms more accurately recover the structure of the generating distribution. In current work, a *constraint-based* algorithm is chosen to construct a probabilistic model of random variables from data.

There have been many *constraint-based* algorithms developed in recent years. Here we select the PC algorithm which has been widely used and is easy to implement [81, 78]. There is no doubt that the method we developed can be generalized to any other BN structure learning algorithm for continuous random variables. The PC algorithm starts by forming a complete undirected graph in

which each pair of nodes are connected by an undirected edge (see Fig. 3.1(b)). Then independence test of any pair of neighboring nodes conditioned on a set of other nodes (called conditional set) are performed. The cardinality of the conditional set is denoted by the order of CI relation. We thins the initial complete undirected graph by removing edges with zero order CI relations, thins again with first order CI relations, and so on until the maximum order is reached. As a result, a graph is constructed from data in a hierarchical way. The complexity of the PC algorithm can be reduced to polynomial complexity by fixing the maximal number of parents of a node [102]. This algorithm is summarized as follows. More details can be found in [81].

- Start with a complete undirected graph $G = (V, E)$ where V is the node set and E is the edge set.
- Set conditional independence test order $n = 0$.

repeat
  for $Y \in V$ **do**
    for $Z \in Adjacent(Y)$ **do**
      for $S \subseteq Adjacent(Y) \backslash \{Z\}$ and $|S| = n$ **do**
        if $Y \perp Z|S$ **then**
          remove edge that connects Y and Z from E, and update the undirected graph G.
        end if
      end for
    end for
  end for
  $n = n + 1$

10: **until** $|Adjacent(Y)\backslash\{Z\}| < n$ or $n = n_{max}$

This algorithm constructs an undirected graphical model (see Fig. 3.2(i)). Then the following criteria are used to form a directed graph to represent the dependence relationships between nodes.

1. For each triple of nodes, e.g. C,E,D such that pair C-E and D-E are adjacent but the pair C,D are not, orient C-E-D as $C \rightarrow E \leftarrow D$ if and only if there is no subset S of $\{E\} \cup V\backslash\{C, D\}$ that D-separates C and D.
2. If $A \rightarrow B$, B and C are adjacent, A and C are not adjacent, and there is no arrowhead at B, then orient B-C as $B \rightarrow C$.
3. If there is a directed path from A to B, and an edge between A and B, then orient A-B as $A \rightarrow B$.

3.2.3 Conditional Independence test

In Bayesian network structure learning, each local CI test is made between two random variables Y and Z given a set of controlling variables $\boldsymbol{X} \in \mathbb{R}^d$. Our problem is to test the hypothesis $Y \perp Z|\boldsymbol{X}$ with N independent data points $\{\boldsymbol{x}_i, y_i, z_i\}_{i=1}^N$. Generally, little prior information is available for the joint distribution of random variables. Hence a nonparametric test is employed to avoid the risk of obtaining incorrect conclusions resulted from parametric modeling. Popular CI testing methods for continuous random variables include linear correlation, Fisher's Z and mutual information. However, they are based on the assumption of multivariate normal data. These measurements are functions of partial correlation coefficients $\rho_{YZ|\boldsymbol{X}}$ which can be estimated from standard

correlation coefficients [47]. For example, when $\boldsymbol{X}$ is a single variable, the expression of partial correlation coefficient reduces to

$$\rho_{YZ|X} = \frac{\rho_{YZ} - \rho_{YX}\rho_{ZX}}{\sqrt{1-\rho_{YX}^2}\sqrt{1-\rho_{ZX}^2}} \tag{3.2}$$

where ρ_{YZ}, ρ_{YX} and ρ_{ZX} are standard correlation coefficients. Since the correlation coefficient of uncorrelated non-Gaussian random variables is always zero even when they are dependent, the methods based on partial correlation coefficients fail to test the conditional independence relationships in such cases.

In this work, we use a nonparametric CI test proposed in [10] which is capable of capturing dependence structure among uncorrelated random variables. The null hypothesis is written as

$$\Pr\{p(Y|Z,\boldsymbol{X}) = p(Y|\boldsymbol{X})\} = 1 \tag{3.3}$$

Given observation data $\{\boldsymbol{x}_i, y_i, z_i\}_{i=1}^N$, we need to accept or reject the hypothesis according to certain statistics. Obviously, this involves exploring the dependence relationships between random variables. To explicitly introduce the dependence structure among $\{\boldsymbol{X}, Y, Z\}$ into the problem, the copulas are employed [64]. A copula is a kind of distribution function that describes the dependence between random variables. Let $U = (U_1, \cdots, U_q) \in \mathbb{R}^q$ a random vector with each component U_i has a uniform distribution on $[0,1]$. The joint cumulative distribution function (CDF) is defined by

$$F_U(u_1, \cdots, u_q) = \Pr(U_1 \leq u_1, \cdots, U_q \leq u_q), \quad u_i \in \mathbb{R} \tag{3.4}$$

The restriction of function F_U to the hypercube $[0,1]^q$ is called a copula function, $C : [0,1]^q \to [0,1]$

$$C(u_1, \cdots, u_q) = F_U(u_1, \cdots, u_q) \tag{3.5}$$

According to Sklar's theorem [64], for an arbitrary random vector $\boldsymbol{X} = (X_1, \cdots, X_d) \in \mathbb{R}^d$, there exists a copula C such that

$$C(F_{X_1}(x_1), \cdots, F_{X_d}(x_d)) = F_{\boldsymbol{X}}(F_{X_1}(x_1), \cdots, F_{X_d}(x_d)) \tag{3.6}$$

where $F_{X_i}(x_i)$ is the marginal distribution of X_i. If all marginal CDFs, $F_{X_i}(x_i), i = 1, \cdots, d$ are continuous, the copula is unique. The copula density $c(\cdot)$ is defined as the derivative of the copula function $C(\cdot)$ with respect to each of its arguments:

$$c(u_1, \cdots, u_d) = \frac{\partial^d}{\partial u_1 \cdots \partial u_d} C(u_1, \cdots, u_d) \tag{3.7}$$

Let $\boldsymbol{x} = (x_1, \cdots, x_d)$ and $\bar{F}_{\boldsymbol{X}}(\boldsymbol{x}) \doteq (F_{X_1}(x_1), \cdots, F_{X_d}(x_d))$, the joint PDF of $\boldsymbol{X}$ can be formulated in terms of its marginal densities and copula density

$$p(\boldsymbol{x}) = \prod_{i=1}^{d} p_{X_i}(x_i) \times c(\bar{F}_{\boldsymbol{X}}(\boldsymbol{x})) \tag{3.8}$$

According to the properties of copula functions, the conditional independence, $Y \perp Z | \boldsymbol{X}$, can be reformulated in terms of copulas as

$$c_{\boldsymbol{X}YZ}(\overline{F}_{\boldsymbol{X}}(\boldsymbol{x}), F_Y(y), F_Z(z)) = c_{\boldsymbol{X}Y}(\overline{F}_{\boldsymbol{X}}(\boldsymbol{x}) F_Y(y)) c_{\boldsymbol{X}Z}(\overline{F}_{\boldsymbol{X}}(\boldsymbol{x}), F_Z(z)) \tag{3.9}$$

Then the null hypothesis in Eq. 3.3 is equivalent to

$$\Pr(c_{\boldsymbol{X}YZ} = c_{\boldsymbol{X}Y} c_{\boldsymbol{X}Z}) = 1, \quad \forall\, y \in \mathbb{R} \tag{3.10}$$

in which the CI test is based on the estimation of copula functions.

According to the copula-based null hypothesis, the similarity between $c_{\boldsymbol{X}YZ}$ and $c_{\boldsymbol{X}Y} c_{\boldsymbol{X}Z}$ can be utilized as a measure of conditional independence. To this end, the Hellinger distance is employed as it is a natural tool of quantifying the similarity between two probability distributions. It is defined as

$$H = \int_{[0,1]^{d+2}} (1 - \sqrt{\frac{c_{\boldsymbol{X}Y}(\boldsymbol{x}, y) c_{\boldsymbol{X}Z}(\boldsymbol{x}, z)}{c_{\boldsymbol{X}YZ}(\boldsymbol{x}, y, z)}})^2 dC_{\boldsymbol{X}YZ}(\boldsymbol{x}, y, z) \tag{3.11}$$

which is equal to zero under conditional independence.

In most cases, the copula is intractable and need to be estimated from observation data. Thus a nonparametric estimator, Bernstein density copula estimator, is adopted here due to its flexibility [76]. Denote the marginal CDFs of a data point $\{\boldsymbol{x}_i, y_i, z_i\}$ by

$$g_i = (F_{X_1}(x_{i,1}), \cdots, F_{X_d}(x_{i,d}), F_Y(y_i), F_Z(z_i)) \tag{3.12}$$

This copula estimator is defined by

$$\widehat{c}_{\boldsymbol{X}YZ}(g_1, \cdots, g_{d+2}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{v_1=0}^{k-1}\cdots\sum_{v_{d+2}=0}^{k-1} A_{g_i,v}\prod_{j=1}^{d+2}\mathcal{B}(g_j, v_j+1, k-v_j) \tag{3.13}$$

where k is an integer (bandwidth parameter) and $\mathcal{B}$ is the beta distribution. $A_{g_i,v}$ is an indicator function $A_{g_i,v} = \mathbf{1}\{g_i \in B_v\}$ with

$$B_v = [\frac{v_1}{k}, \frac{v_1+1}{k}] \times \cdots \times [\frac{v_{d+2}}{k}, \frac{v_{d+2}+1}{k}]$$

Recall that the random vector $\boldsymbol{X} \in \mathbb{R}^d$. The total number of random variables in the CI test is $d+2$. All empirical marginal distributions are estimated with conventional kernel density estimators. Then the estimator for Hellinger distance is

$$\widehat{H} \approx \frac{1}{N}\sum_{i=1}^{N}(1 - \sqrt{\frac{\widehat{c}_{\boldsymbol{X}Y}(\bar{F}_{\boldsymbol{X}}(\boldsymbol{x}_i), F_Y(y_i))\widehat{c}_{\boldsymbol{X}Z}(\bar{F}_{\boldsymbol{X}}(\boldsymbol{x}_i), F_Z(z_i))}{\widehat{c}_{\boldsymbol{X}YZ}(\bar{F}_{\boldsymbol{X}}(\boldsymbol{x}_i), F_Y(y_i), F_Z(z_i))}})^2 \tag{3.14}$$

Based on Hellinger's distance H and copula densities, the statistics, T, derived in [10] is used to test the conditional independence.

$$T \equiv \frac{Nk^{-(d+2)/2}}{\sigma}(4H - N^{-1}C_1k^{(d+2)/2} - N^{-1}B_1k^{(d+1)/2} - N^{-1}B_2k^{(d+2)/2} - N^{-1}B_3k^{d/2}) \tag{3.15}$$

where

$$
\begin{aligned}
C_1 &= 2^{-(d+2)}\pi^{(d+2)/2}, \quad \sigma = \sqrt{2}(\pi/4)^{(d+2)/2} \\
B_1 &= -2^{-d}\pi^{(d+1)/2} + \frac{1}{N}\sum_{i=1}^{N} \frac{\prod_{j=1}^{d+1}(4\pi g_{ij}(1-g_{ij}))^{-1/2}}{c_{\boldsymbol{X}Y}(g_{i1},\cdots,g_{i(d+1)})} \\
B_2 &= -2^{-d}\pi^{(d+1)/2} \\
&+ \frac{1}{N}\sum_{i=1}^{N} \frac{4\pi(g_{i(d+2)}(1-g_{i(d+2)}))^{-1/2}\prod_{j=1}^{d}(4\pi g_{ij}(1-g_{ij}))^{-1/2}}{c_{\boldsymbol{X}Z}(g_{i1},\cdots,g_{id},g_{i(d+2)})} \\
B_3 &= 2^{-(d+1)}\pi^{-d/2}\frac{1}{N}\sum_{i=1}^{N} \frac{c_{\boldsymbol{X}}(g_{i1},\cdots,g_{id})}{\sqrt{\prod_{j=1}^{d} g_{ij}(1-g_{ij})}}
\end{aligned}
\tag{3.16}
$$

where g_{ij} is the j-th element of the vector g_i in Eq (3.12). This test statistic is asymptotically normal with weak assumptions under null hypothesis [10], i.e. $T \sim \mathcal{N}(0,1)$. After obtaining numerical value of T from observation data, we can compute the probability of observation data under null hypothesis (i.e. the p-value) and compare it with predefined significance level α to determine whether rejecting the hypothesis of conditional independence relationship between Y and Z conditioned on $\boldsymbol{X}$.

Since the observation data are generally limited in practice, the following local smoothed bootstrap algorithm is employed to do resampling from observation data to compute the p-value.

1. Given observation data $\{\boldsymbol{x}_i, y_i, z_i\}_{i=1}^{N}$, calculate T using equation (3.15).
2. Draw a sample $\boldsymbol{x}^{(r)}$ from a conventional kernel density estimator
3. Conditioned on $\boldsymbol{x}^{(r)}$, we draw $y^{(r)}$ and $z^{(r)}$ independently from the conditional density estimator for $p(y|\boldsymbol{x})$ and $p(z|\boldsymbol{x})$
4. Compute corresponding Hellinger distance $H^{(r)}$ and the test statistic $T^{(r)}$ using Eqs (3.11) and (3.15)

5. Repeat steps (2)-(4) R times to get a set of values $\{T^{(r)}\}_{r=1}^{R}$. The bootstrap p-value is
$$p = \frac{1}{R}\sum_{r=1}^{R}\mathbf{1}(T^{(r)} > T)$$
where $\mathbf{1}(\cdot)$ is an indicator function.

6. Given a significance level α (typically 0.05), we reject the conditional independence assumption if $p < \alpha$.

3.3 Gaussian mixture modeling of conditional distributions

With the help of graphical model, the multivariate distribution $p(\boldsymbol{\eta})$ is decomposed into products of lower-dimensional conditional distributions. Then it is necessary to estimate the conditional distributions from observation data. In this section, we adopt the Gaussian mixture modeling for nonparametric density estimation.

Consider a conditional distribution $p(\eta|\tilde{\eta})$ where η is a single random variable and $\tilde{\eta}$ is another variable or vector. Let $\bar{\eta} \doteq (\eta, \tilde{\eta}^T)^T$. Assume that the joint distribution $p(\bar{\eta})$ can be accurately approximated by a multivariate Gaussian mixture, i.e.
$$p(\bar{\eta}) \approx \sum_{i=1}^{t} \bar{w}_i \mathcal{N}(\bar{\eta}; \bar{\mu}_i, \bar{\Sigma}_i) \tag{3.17}$$
When $\dim(\tilde{\eta})$ is relatively small, an expectation-maximization (EM) algorithm can be utilized to learn the weights, means and covariances from observation data [7].

According to Bayes' rule, the conditional distribution
$$p(\eta|\tilde{\eta}) = \frac{p(\eta, \tilde{\eta})}{p(\tilde{\eta})} \propto \sum_{i=1}^{r} W_i(\tilde{\eta})\mathcal{N}(\eta; \mu_i(\tilde{\eta}), \sigma_i^2(\tilde{\eta})) \tag{3.18}$$

Recall Eq 3.17, we denote by $\bar{\mu}_i^T = (\mu_i', \tilde{\mu}_i^T)$ and $\bar{\Gamma}_i = (\bar{\Sigma}_i)^{-1}$ with submatrices $[\bar{\Gamma}_{i,11}]_{1\times 1}$,$[\bar{\Gamma}_{i,12}]_{1\times d}$, $[\bar{\Gamma}_{i,21}]_{d\times 1}$ and $[\bar{\Gamma}_{i,22}]_{d\times d}$. Then it is straightforward to get the mean and variance in each component as

$$\sigma_i^2(\tilde{\eta}) = \frac{1}{\bar{\Gamma}_{i,11}}, \quad \mu_i(\tilde{\eta}) = \mu_i' - \sigma_i^2 \bar{\Gamma}_{i,12}(\tilde{\eta} - \tilde{\mu}_i) \tag{3.19}$$

The unnormalized weights are formulated as

$$W_i(\tilde{\eta}) = \bar{w}_i \exp(-\frac{1}{2} f(\tilde{\eta})) \tag{3.20}$$

such that $f(\tilde{\eta}) = \mu_i' \bar{\Gamma}_{i,11} - \mu_i^2 - 2\mu_i' \bar{\Gamma}_{i,12}(\tilde{\eta} - \tilde{\mu}_i) + (\tilde{\eta} - \tilde{\mu}_i)^T \bar{\Gamma}_{i,22}(\tilde{\eta} - \tilde{\mu}_i)$.

3.4 Stochastic reduced-order modeling via KL expansion

In this section, we combine the graph-based probabilistic modeling with model reduction techniques to construct a reduced-order representation of high-dimensional random field from observation data. One of the most popular model reduction methods are Karhunen-Loève (KL) decomposition and its variants. Denote the correlation function by $C(\boldsymbol{x}, \boldsymbol{x}')$ where $\boldsymbol{x}$ are coordinates of a point in the domain. The KL expansion takes the following form:

$$\boldsymbol{a}(\boldsymbol{x}, \omega) = \mathbb{E}[\boldsymbol{a}(\boldsymbol{x})] + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \phi_i(\boldsymbol{x}) \eta_i(\omega) \tag{3.21}$$

where $\{\lambda_i, \phi_i(\boldsymbol{x})\}_{i=1}^{\infty}$ are eigenpairs of the correlation function

$$\int_D C(\boldsymbol{x}, \boldsymbol{x}') \phi_i(\boldsymbol{x}') d\boldsymbol{x}' = \lambda_i \phi_i(\boldsymbol{x}) \tag{3.22}$$

The KL expansion coefficients $\{\eta_i\}_{i=1}^{\infty}$ satisfy

$$\eta_i = \frac{1}{\sqrt{\lambda_i}} \int_D (\boldsymbol{a}(\boldsymbol{x}, \omega) - \mathbb{E}[\boldsymbol{a}(\boldsymbol{x})]) \phi_i(\boldsymbol{x}) d\boldsymbol{x} \tag{3.23}$$

Accordingly, they have the following properties:

$$\mathbb{E}[\eta_i] = 0, \quad \mathbb{E}[\eta_i \eta_j] = \delta_{ij} \tag{3.24}$$

where δ_{ij} is the Kronecker delta.

Generally, the eigenvalue problem in Eq (3.22) does not have analytic solutions. Therefore, numerical methods are adopted in many practical applications. Assume we are given a set of independent realizations of the random field, $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N\}$ where each realization $\boldsymbol{a}_i \in \mathbb{R}^p$ is a column vector. The eigenpairs in the KL expansion can be obtained directly from an unbiased estimator of the underlying covariance matrix $\hat{C}$

$$\hat{C} = \frac{1}{N-1} \sum_{k=1}^{N} (\boldsymbol{a}_k - \overline{\boldsymbol{a}})(\boldsymbol{a}_k - \overline{\boldsymbol{a}})^T \tag{3.25}$$

where $\overline{\boldsymbol{a}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{a}_i$. The associated realizations of random variables $\{\eta_i\}_{i=1}^{\infty}$ are given by

$$\eta_{i,t} = \frac{1}{\sqrt{\lambda_i}} (\boldsymbol{a}_t - \overline{\boldsymbol{a}})^T \phi_i, \quad t = 1, \cdots, N \tag{3.26}$$

In practice, the KL representation in Eq (3.38) is truncated by by taking the first q terms

$$\boldsymbol{a}(\boldsymbol{x}, \omega) \approx \boldsymbol{a}_q(\boldsymbol{x}, \omega) = \overline{\boldsymbol{a}} + \sum_{i=1}^{q} \sqrt{\lambda_i} \phi_i(\boldsymbol{x}) \eta_i(\omega) \tag{3.27}$$

Thus the original random field is represented in terms of m zero-mean uncorrelated random variables $\boldsymbol{\eta} \doteq \{\eta_1, \cdots, \eta_q\}$.

The next problem is to represent the random vector $\boldsymbol{\eta}$ based on independent samples $\{\boldsymbol{\eta}_1, \cdots, \boldsymbol{\eta}_N\}$. According to Eq (3.24), the KL expansion coefficients are uncorrelated. For a Gaussian random field, $\boldsymbol{\eta}$ are in a multivariate Gaussian distribution and pairwise uncorrelatedness implies independence. The estimation of joint distribution reduces to a number of trivial one-dimensional density estimation. However, this is not the case for arbitrary stochastic processes. The KL

expansion coefficients can be dependent and have nonstandard distributions determined by data. Therefore, the graph-based factorization of the joint distribution $p(\boldsymbol{\eta})$ proposed in section 3.2 are employed. Then the lower-dimensional conditional distributions are estimated by Gaussian mixture models. In this way, it is straightforward to make samples of these KL expansion coefficients from the graphical model and use KL expansion in Eq 3.27 to generate realizations of the target random field.

3.4.1 Polynomial Chaos representation of the reduced-order model

In previous sections, we construct a framework for generating independent realizations of random field. However, in many applications, it is desirable to map the random vector $\boldsymbol{\eta}$ to a set of independent identically distributed random variables $\boldsymbol{\xi}$. It makes it easy to apply methods other than MC simulation, e.g. collocation methods, for uncertainty quantification.

The most commonly used approach for this purpose is polynomial chaos (PC) expansion. In this approach, any random variable with finite variance can be expanded in terms of orthogonal polynomials of specific standard random variables [27, 97, 98]. A pth order PC expansion of the random vector $\boldsymbol{\eta}$ can be expressed as

$$\boldsymbol{\eta} = \sum_{|\boldsymbol{\alpha}|=0}^{p} c_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \tag{3.28}$$

where $\boldsymbol{\xi} = \{\xi_1, \cdots, \xi_q\}$ is a vector of standard random variables and $\boldsymbol{\alpha} = \{\alpha_1, \cdots, \alpha_q\} \in \mathbb{N}^q$ is a multi-index with modulus $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_q$. The

multivariate polynomial basis function $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})$ is defined as

$$\Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \psi_{\alpha_1}(\xi_1) \cdots \psi_{\alpha_q}(\xi_q) \tag{3.29}$$

where $\psi_{\alpha_i}(\cdot)$ is the standard 1D polynomial of degree α_i. In this work, we use Hermite polynomials. The standard 1D Hermite polynomials are defined by

$$\begin{aligned} &\Psi_0(\xi_i) = 1, \quad \Psi_1(\xi_i) = \xi_i \\ &\Psi_{j+1}(\xi_i) = \xi_i \Psi_j(\xi_i) - j\Psi_{j-1}(\xi_i), \quad \text{when} \quad j > 1 \end{aligned} \tag{3.30}$$

These polynomials are orthogonal with respect to the corresponding probability density function of standard normal random variables:

$$\mathbb{E}[\Psi_i \Psi_j] = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \Psi_i(\eta)\Psi_j(\eta) \exp(-\frac{\eta^2}{2}) d\eta = i!\delta_{ij} \tag{3.31}$$

Due to the orthogonality of polynomials, the Galerkin projection is used to calculate the PC coefficients:

$$c_{\boldsymbol{\alpha}} = \frac{\mathbb{E}[\boldsymbol{\eta}\Psi_\alpha(\boldsymbol{\xi})]}{\mathbb{E}[\Psi^2_{\boldsymbol{\alpha}}(\boldsymbol{\xi})]} = \frac{1}{\mathbb{E}[\Psi^2_{\boldsymbol{\alpha}}(\boldsymbol{\xi})]} \int \boldsymbol{\eta}\Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) p(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{3.32}$$

where $p(\boldsymbol{\xi})$ is the joint probability of $\boldsymbol{\xi}$. As the polynomial is determined, the $\boldsymbol{\xi}$ is subject to a standard distribution with an analytic expression. However, the KL expansion coefficients $\boldsymbol{\eta}$ does not belong to the same stochastic space as $\boldsymbol{\xi}$. To compute this integral, a mapping $\Gamma : \boldsymbol{\xi} \to \boldsymbol{\eta}$ is necessary such that $\Gamma(\boldsymbol{\xi})$ and $\boldsymbol{\eta}$ have the same distributions.

To compute the PC coefficients with Eq (3.32), we employ the Rosenblatt transformation in conjunction with the probabilistic model of $\boldsymbol{\eta}$ obtained from graphical model in section 3.2:

$$\xi_i = F_G^{-1} \circ F(\eta_i | \Pi_{\eta_i}) \tag{3.33}$$

where $F_G(\cdot)$ is the CDF of a standard Gaussian random variable and $F(\cdot|\cdot)$ indicates a conditional cumulative distribution. Recall that Π_{η_i} denote the parent

nodes of η_i in a Bayesian network. To obtain the mapping,$\Gamma : \boldsymbol{\xi} \rightarrow \boldsymbol{\eta}$, an inverse of the Rosenblatt transformation in Eq (3.33) is taken. The empirical conditional CDF evaluated here is monotonic, which ensures the inverse transformation can be performed.

3.5 Numerical example

In this section, we apply the proposed approach to model a non-Gaussian random field. The generated stochastic input model is used as an input to a single phase incompressible fluid flow in porous media. The governing equations are [1]

$$\nabla \cdot \boldsymbol{u} = f, \ \boldsymbol{u} = -K(\boldsymbol{x}, \omega)\nabla p, \quad \forall \boldsymbol{x} \in D, \tag{3.34}$$

with boundary conditions

$$p = \overline{p} \text{ on } \partial D_p, \quad \boldsymbol{u} \cdot \boldsymbol{n} = \overline{\boldsymbol{u}} \text{ on } \partial D_u, \tag{3.35}$$

with the assumptions that the effects of gravity, capillary pressure, and compressibility can be neglected and that the porosity is constant. The source term in Eq. (4.3) is used to model injection/production wells:

$$f(\boldsymbol{x}) = \begin{cases} -r, & \text{if } 0 \leq x_i < w, \text{ for } i = 1,2, \\ r, & \text{if } 1 - w \leq x_i < 1, \text{ for } i = 1,2, \\ 0 & \text{otherwise.} \end{cases} \tag{3.36}$$

The parameters are chosen to be $r = 1$ and $w = 1/64$. No-flow homogeneous Neumann boundary conditions are applied on all boundaries. Standard mixed finite element method is adopted to solve the equations [70].

The spatial domain is a unit square $[0, 1]^2$ which is discretized into a 64×64 grid. The permeability is defined as a constant in each grid block. For simplicity,

we let the random permeability tensor $K(\boldsymbol{x}, \omega)$ be isotropic and use the log-permeability $\boldsymbol{a}(\boldsymbol{x}, \omega) = \log(K)$ as the underlying stochastic input. $\boldsymbol{a}(\boldsymbol{x}, \omega)$ is assumed to be a Gaussian random field with mean zero and an exponential covariance function defined as

$$\operatorname{cov}(\boldsymbol{x}, \boldsymbol{x}^\star) = \sigma^2 \exp(-\frac{|x_1 - x_1^\star|}{L_1} - \frac{|x_2 - x_2^\star|}{L_2}), \tag{3.37}$$

where coordinates $\boldsymbol{x} = (x_1, x_2)$ and $\boldsymbol{x}^\star = (x_1^\star, x_2^\star)$ and σ is the standard deviation of the random field. An isotropic random field is assumed such that correlation lengths $L_1 = L_2 = 0.1$ and the standard deviation $\sigma = 1.5$. The samples of permeability are generated using standard KL expansion with the first 10 terms, i.e.

$$\boldsymbol{a}(\boldsymbol{x}, \omega) = \sum_{i=1}^{10} \sqrt{\lambda_i} \phi_i(\boldsymbol{x}) \eta_i \tag{3.38}$$

where $\{\lambda_1, \cdots, \lambda_{10}\}$ are the largest eigenvalues arranged in descending order. A non-Gaussian multivariate distribution is specified for KL expansion coefficients $\boldsymbol{\eta} \doteq \{\eta_1, \cdots, \eta_q\}$ where $q = 10$:

$$\begin{aligned}
\eta_2 &\propto \tilde{\eta}_2, \quad \tilde{\eta}_2 \sim 0.4\mathcal{N}(-6, 4) + 0.6\mathcal{N}(4, 1) \\
\eta_1 &\propto \tilde{\eta}_2 X, \quad X \sim 0.2\mathcal{N}(8.0.25) + 0.8\mathcal{N}(-2, 0.25) \\
\eta_3 &\propto \tilde{\eta}_2 Y, \quad Y \sim 0.5\mathcal{N}(-10, 1) + 0.5\mathcal{N}(10, 1) \\
\eta_i &\propto \tilde{\eta}_i, \quad \tilde{\eta}_i \sim 0.4\mathcal{N}(-6, 4) + 0.6\mathcal{N}(4, 1) \\
\eta_{i+1} &\propto \tilde{\eta}_i X, \quad \text{for } i = 4, 6, 8
\end{aligned} \tag{3.39}$$

and $\eta_{10} \sim \mathcal{N}(0, 1)$. These random variables are all standardized such that $\mathbb{E}(\eta_i) = 0$ and $\mathbb{E}(\eta_i^2) = 1$.

Monte Carlo (MC) simulation is conducted with 10^5 realizations directly sampled from the random field defined in Eqs. (3.38) and (3.39). The statistics

obtained in this way are taken as reference. To evaluate the performance of the proposed algorithm, $N = 2000$ realizations, $\mathcal{D} = \{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N\}$, are randomly picked up as training data. Two KL expansion based methods are employed to construct stochastic input models. In the first one, we assume that the KL expansion coefficients are mutually independent; while in the second one, the joint distribution of these coefficients are explored by Bayesian network structure learning. MC simulations are conducted with both input models.

3.5.1 Approximation of the joint distribution of KL expansion coefficients

Given the training data of stochastic input, samples of coefficients $\boldsymbol{\eta}$, i.e. $\mathcal{D}_\eta = \{\boldsymbol{\eta}_1, \cdots, \boldsymbol{\eta}_N\}$, can be obtained directly from KL expansion. We learn the dependence structure among these variables and factorize the joint PDF $p(\boldsymbol{\eta})$ into lower dimensional conditional PDFs using the algorithm proposed in section 3.2.2. According to Eq. (3.39), the joint distribution can be written as

$$p(\boldsymbol{\eta}) = p(\eta_2)p(\eta_1|\eta_2)p(\eta_3|\eta_2)p(\eta_{10}) \prod_{i\in\{4,6,8\}} p(\eta_i)p(\eta_{i+1}|\eta_i) \tag{3.40}$$

The corresponding graphical model is depicted in Fig. 3.1(a). As discussed in section 3.2.2, a natural initial guess of the dependence structure is a fully connected graph shown in Fig. 3.1(b), which implies that any set of variables are mutually dependent. However, the real probabilistic model may not be as complex as this owing to the existence of conditional independence. By running local CI tests, the initial graphical model is thinned by removing redundant edges. The significance level α in the CI test is set to be 5% in this example. Some in-

termediate graphical models in the learning process are presented in Fig. 3.2. Fig. 3.2(b) shows the graphical model after running the first order CI test with respect to variable η_1. According to the true dependence relationship expressed in Eq. (3.40), η_1 only depends on η_2. The conditional independence removes the edges between η_1 and all the other nodes except η_2. In Fig. 3.2(e), a stand-alone cluster, $\{\eta_1, \eta_2, \eta_3\}$, is found. Finally, we obtain an undirected graphical model corresponding to the true probabilistic model by removing all redundant edges between conditionally independent random variables. Then the rules in section 3.2.2 convert the undirected graph into a Bayesian network. Note that these rules may lead to incorrect directions of arrows and may leave the directions of some edges undefined. For example, according to the CI tests, we have $p(\eta_1, \eta_3|\eta_2) = p(\eta_1|\eta_2)p(\eta_3|\eta_2)$. However, there are three possible graph structures satisfying this conditional independence relationship, (1) $\eta_1 \leftarrow \eta_2 \rightarrow \eta_3$, (2) $\eta_1 \rightarrow \eta_2 \rightarrow \eta_3$ and (3) $\eta_1 \leftarrow \eta_2 \leftarrow \eta_3$. They correspond to different but equivalent factorization of the joint distribution $p(\eta_1, \eta_2, \eta_3)$, i.e.

$$\begin{aligned} p(\eta_1, \eta_2, \eta_3) &= p(\eta_1|\eta_2)p(\eta_3|\eta_2)p(\eta_2) \\ &= p(\eta_1)p(\eta_2|\eta_1)p(\eta_3|\eta_2) \\ &= p(\eta_3)p(\eta_2|\eta_3)p(\eta_1|\eta_2) \end{aligned} \tag{3.41}$$

As Gaussian mixture model is utilized to approximate the joint distribution, the three factorizations can give almost the same results except slight difference resulted from numerical errors. In this work, we assume that KL coefficients with small indices dominate those with larger indices in order to determine undefined directions of edges. The final Bayesian network is shown in Fig. 3.3.

Given the factorized joint PDF of $\boldsymbol{\eta}$, it is straightforward to approximate the

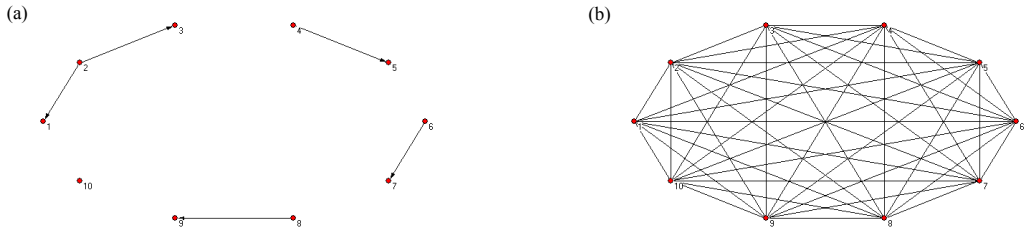

Figure 3.1: (a) True graphical model for random variables $\boldsymbol{\eta}$, and (b) the initial guess of the dependence structure

conditional distributions with Gaussian mixture models. In this example, we use Gaussian mixtures with 8 components. In Fig. 3.4, we compare the samples of η_1 and η_2 from their marginal distributions with the samples from conditional Gaussian mixture distributions. The simulated experiment data are also presented for reference. In Fig. 3.5, we present the samples of η_2 and η_3. In both cases, the variables are dependent. As a result, sampling from marginal distributions leads to incorrectly distributed samples of KL expansion coefficients, which will further affect the distribution of samples of stochastic input.

3.5.2 Uncertainty propagation with the stochastic input model

Besides the graph-based stochastic input model, we also construct a stochastic input model by assuming that the KL expansion coefficients are mutually independent for comparison. The Gaussian mixture model is applied to approximate the PDF of each coefficient. 10^5 samples are generated from both stochastic input models respectively. Note that in both input models, the Gaussian mixture modeling is performed with 4 components. The contour plots of variance of pressure and velocities are given in Fig. 3.6. From this figure, it is seen that the variance of model responses obtained from both stochastic input models are

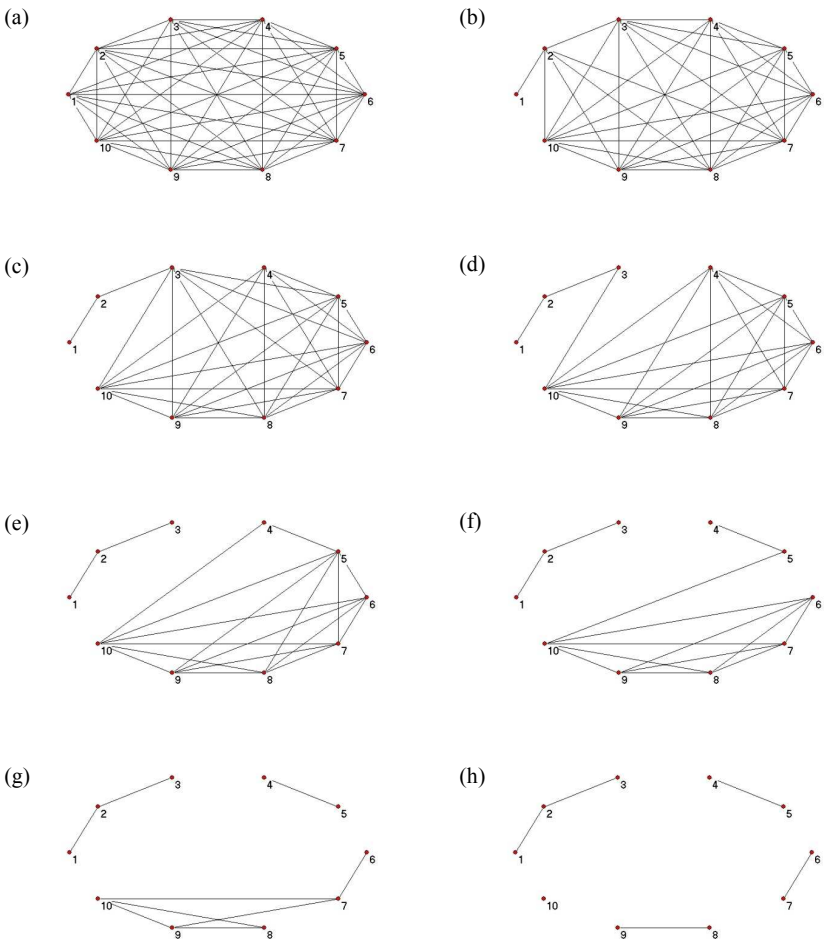

Figure 3.2: (a) Initial graph structure, (b)-(g) intermediate graphical models during dependence structure learning process and (h) final undirected graph structure

nearly the same with reference solutions. In other words, the variance can be accurately estimated by capturing marginal PDFs of $\boldsymbol{\eta}$.

Obviously, only consider first- and second-order statistics may not be enough in many cases. Therefore we examine high-order statistics, skewness and kurtosis, in this work. Given N samples, $\{X_1, \cdots, X_N\}$, of a random variable X, the skewness is defined as

$$\text{skewness} = \sum_{i=1}^{N} (X_i - \bar{X})^3 / (N-1) s^3 \tag{3.42}$$

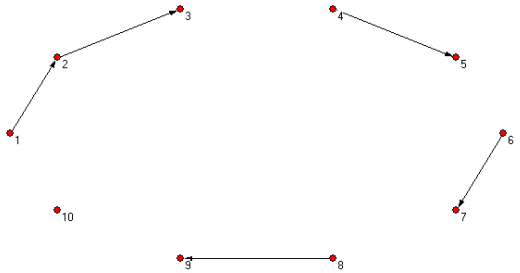

Figure 3.3: Final Bayesian network converted from the undirected graph in Fig. 3.2(h).

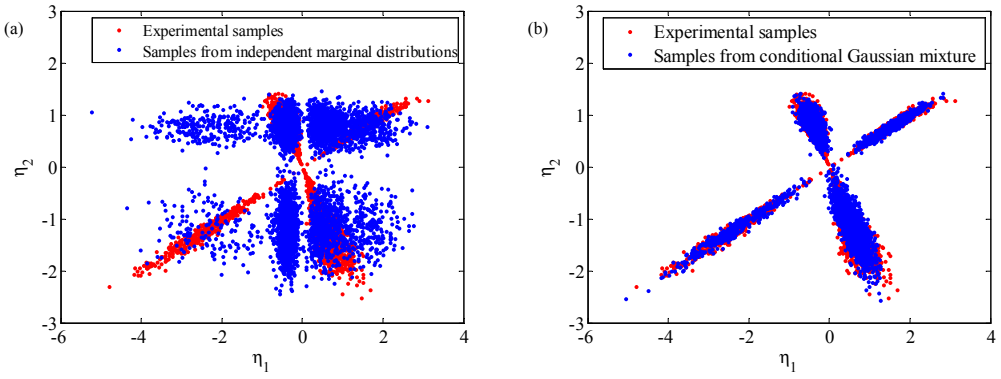

Figure 3.4: Samples of η_1 and η_2 obtained from (a) their marginal distributions, and from (b) conditional Gaussian mixture distributions.

and the kurtosis is defined as

$$\text{kurtosis} = \sum_{i=1}^{N} (X_i - \bar{X}^4)^4 / (N-1)s^4 \tag{3.43}$$

where s is the standard deviation of X and $\bar{X}$ denotes the mean value. The results are given in Fig. 3.7 and 3.8 respectively. We can see that the graph-based stochastic input model gives more accurate predictions of high-order statistics. For example, if the dependence relationships between KL expansion coefficients are missing, the estimated contour plots of skewness and kurtosis of pressure exhibit high symmetry, which leads to a wrong conclusion.

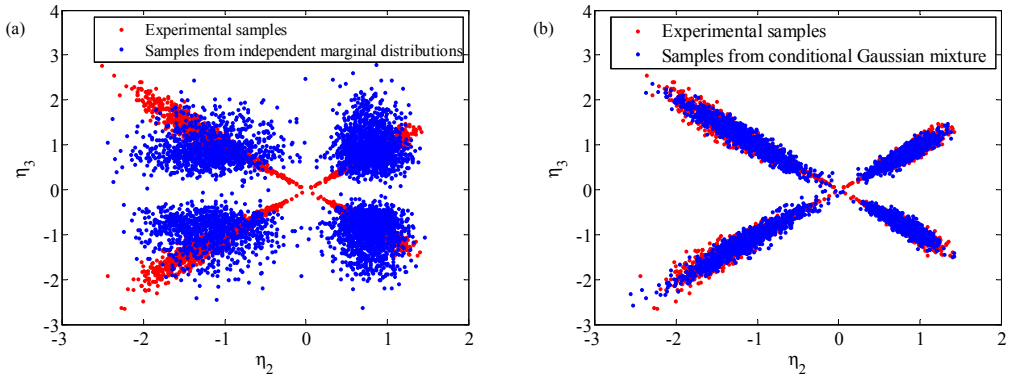

Figure 3.5: Samples of η_2 and η_3 obtained from (a) their marginal distributions, and from (b) conditional Gaussian mixture distributions.

3.6 Conclusions

In this chapter, we proposed a new stochastic reduced-order modeling technique based on Bayesian network structure learning algorithm. The truncated KL expansion has been employed to represent a random field in terms of finite random variables $\boldsymbol{\eta}$. Then the Bayesian network is introduced to evaluate the joint probability of these variables. By running a number of local CI tests, the probabilistic model of $\boldsymbol{\eta}$ can be constructed with a BN structure learning algorithm such that the dependence relations between the random variables are preserved. Since the most common conditional independence tests for continuous random variables are based on correlation coefficient which is unable to explore the dependence between uncorrelated KL random variables, a nonparametric copula based CI test is employed. As a result, the multivariate PDF is decomposed into a set of conditional distributions based on the dependence according to the structure of corresponding BN. Due to the existence of conditional independence, a random variable could depend only on a few other ones rather than its complementary set. Consequently, these conditional distributions can

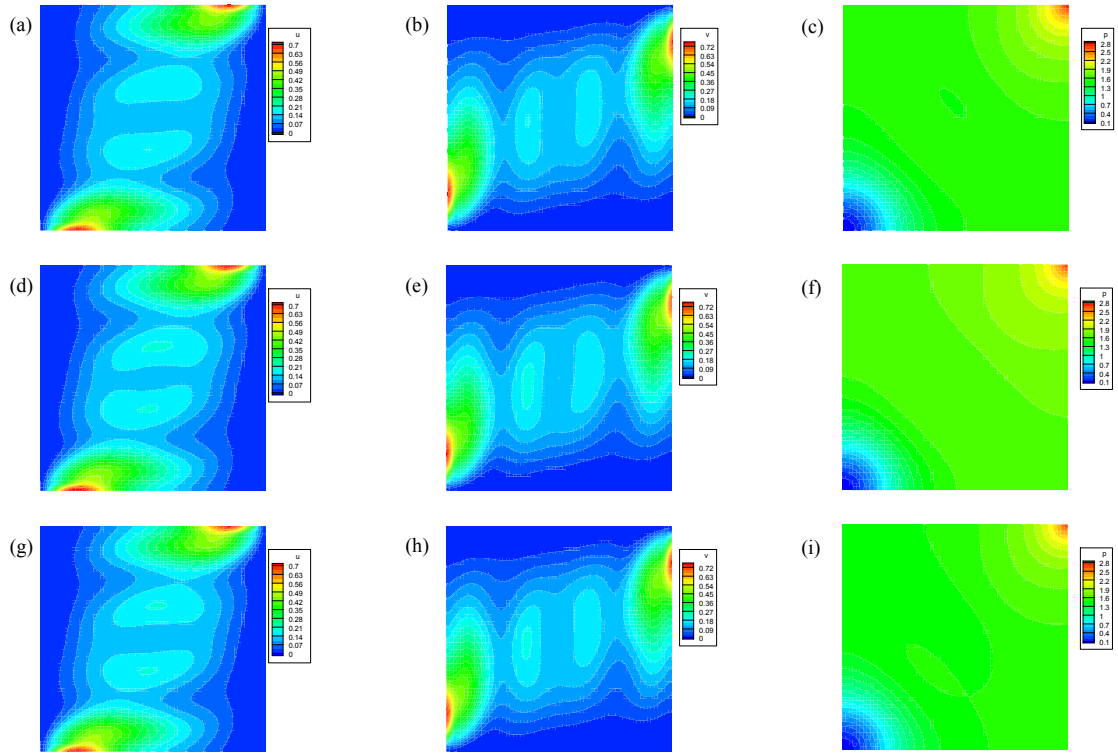

Figure 3.6: Contours of variance of x-velocity (left), y-velocity (middle) and pressure (right) from (a)-(c) reference solutions, (d)-(f) stochastic input model with independent KL expansion coefficients, and (g)-(i) stochastic input model with $p(\boldsymbol{\eta})$ approximated by graphical model.

be separately estimated under low dimensions. Finally, PC expansion is used to represent the KL random variables in terms of standard random variables based on the Rosenblatt transformation.

This work actually develops a general framework of constructing stochastic input models with BN structure learning. The structure learning algorithm and empirical density estimation are not limited to the approaches used. In recent years, more and more advanced constraint-based BN structure learning algorithms have been developed in machine learning and artificial intelligence [102, 82, 15]. It is quite straightforward to replace the structure learning

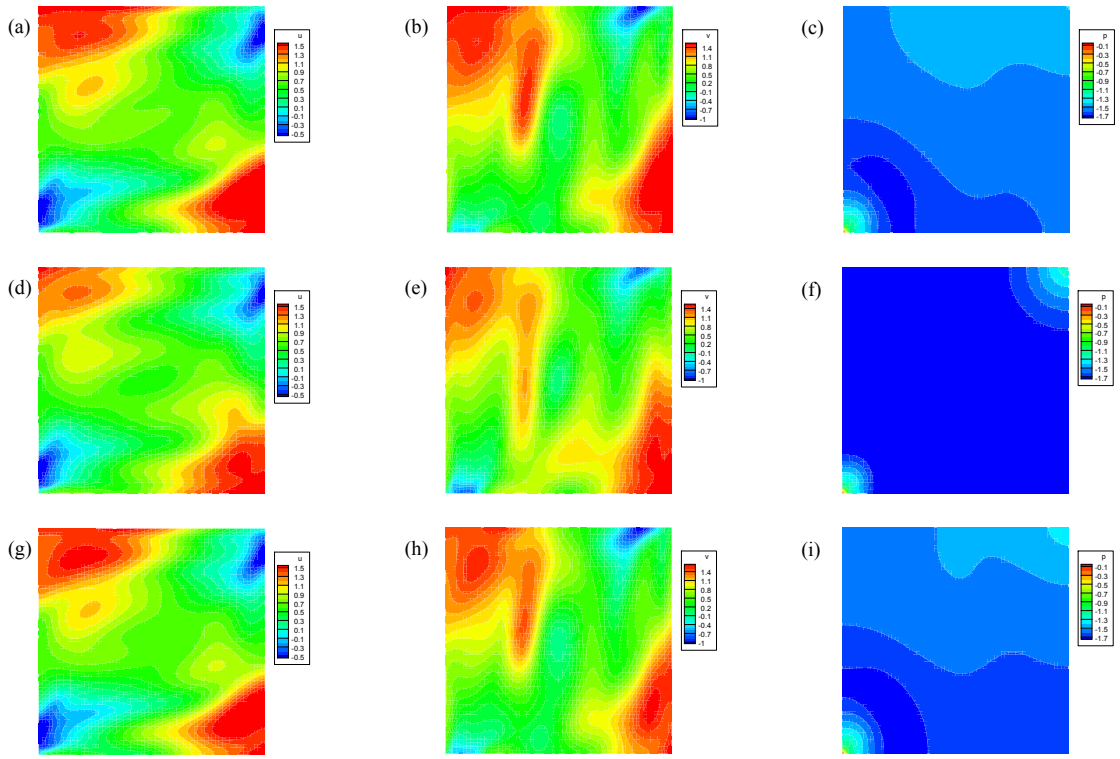

Figure 3.7: Contours of skewness of x-velocity (left), y-velocity (middle) and pressure (right) from (a)-(c) reference solutions, (d)-(f) stochastic input model with independent KL expansion coefficients, and (g)-(i) stochastic input model with $p(\boldsymbol{\eta})$ approximated by graphical model.

algorithm used in this work with any other one to achieve the best accuracy and performance in practical applications.

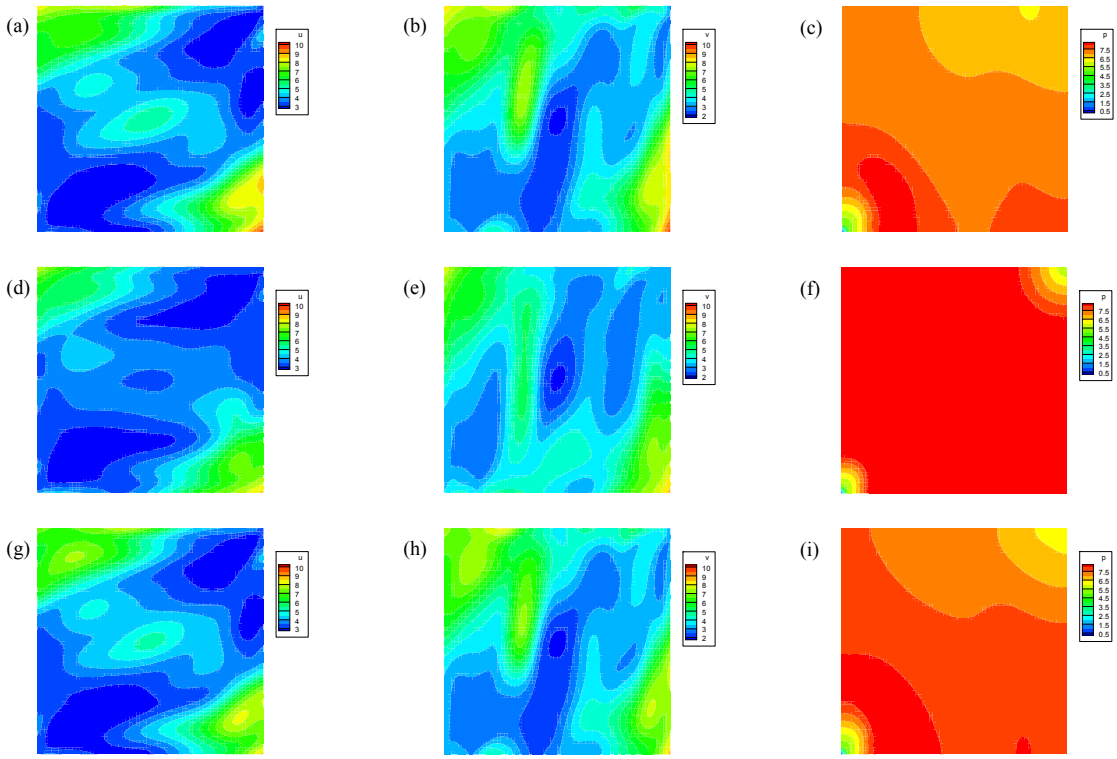

Figure 3.8: Contours of kurtosis of x-velocity (left), y-velocity (middle) and pressure (right) from (a)-(c) reference solutions, (d)-(f) stochastic input model with independent KL expansion coefficients, and (g)-(i) stochastic input model with $p(\boldsymbol{\eta})$ approximated by graphical model.

CHAPTER 4

SOLVING STOCHASTIC MULTISCALE PARTIAL DIFFERENTIAL EQUATIONS: A PROBABILISTIC GRAPHICAL MODEL APPROACH

The model reduction techniques give a lower-dimensional representation of high-dimensional stochastic input. However, in many applications, a reduced dimensionality of input is still large (e.g. several tens or hundreds of dimensions) for conventional sampling based approaches for uncertainty quantification. In this chapter, we develop a probabilistic graphical model based methodology for uncertainty quantification in the presence of stochastic input and multiple scales. Both the stochastic input and model responses are treated as random variables in this framework. Their relationships are modeled by graphical models which give explicit factorization of a high-dimensional joint probability distribution.The hyperparameters in the probabilistic model are learned using SMC method, which is superior to standard MCMC methods for multi-modal distributions. The predictions from the probabilistic graphical model are conducted using belief propagation algorithms. This chapter closely follows the work in [89].

4.1 Problem definition

To model uncertainties in a physical system, we define a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ where Ω is a sample space, $\mathcal{F}$ a σ-algebra of subsects of Ω and $\mathcal{P} : \mathcal{F} \to [0, 1]$ a probability measure. Let $D \subset \mathbb{R}^d$ be a fixed d-dimensional bounded domain with boundary ∂D. A general stochastic partial differential

equation (SPDE) is formulated as

$$\mathcal{L}(\boldsymbol{x}, \omega; y) = 0, \ \forall \boldsymbol{x} \in D, \tag{4.1}$$

with boundary conditions

$$\mathcal{B}(\boldsymbol{x}, \omega; y) = 0, \ \forall \boldsymbol{x} \in \partial D, \tag{4.2}$$

where $\mathcal{L}$ is a general differential operator, $\mathcal{B}$ is a boundary operator, $\omega \in \Omega$ is an elementary event in the sample space and y is the model response.

We are interested in assessing macroscopic quantities from fine-scale information based on SPDEs with multiscale features. For demonstration of the approach, we consider single phase incompressible fluid flow in porous media where the length scale of permeability variation is orders of magnitude smaller than the characteristic length scale of the domain. The pressure h and velocity $\boldsymbol{u}$ are characterized by the following equations [2]

$$\nabla \cdot \boldsymbol{u} = f, \ \boldsymbol{u} = -K(\boldsymbol{x}, \omega)\nabla h, \quad \forall \boldsymbol{x} \in D, \tag{4.3}$$

with boundary conditions

$$h = \overline{h} \text{ on } \partial D_h, \quad \boldsymbol{u} \cdot \boldsymbol{n} = \overline{\boldsymbol{u}} \text{ on } \partial D_u, \tag{4.4}$$

with the assumptions that the effects of gravity, capillary pressure, and compressibility can be neglected and that the porosity is constant.

For simplicity, we let the random permeability tensor $K(\boldsymbol{x}, \omega)$ be isotropic and use the log-permeability $a(\boldsymbol{x}, \omega) = \log(K)$ as it is commonly employed in geostatistical models.

Due to the multiscale nature of the problem, two sets of grids are defined. First, let us discretize the domain D into non-overlapping elements e_i and obtain

a fine-scale grid $\mathcal{T}_h = \bigcup_{i=1}^{N_h} e_i$, where N_h is the number of fine elements. In this work, the global log-permeability is defined as $\boldsymbol{a} \equiv \{a_i(\omega)\}_{i=1}^{N_h}$ where $a_i(\omega)$ is the log-permeability on the i-th fine-scale element. Then we define the skeleton of the fine-scale partition, $\mathcal{P}_h = \bigcup_{a=1}^{M_h} \nu_a$, where ν_a denote element faces and M_h is the total number of faces. In a multiscale problem, a coarse-scale partition of the same domain is proposed. Denote this partition as $\mathcal{T}_c = \bigcup_{i=1}^{N_c} E_i$ where E_i are coarse elements. We also denote the associated skeleton of the coarse-scale discretization by $\mathcal{P}_c = \bigcup_{a=1}^{M_c} \Lambda_a$. Here, N_c is the number of coarse elements and M_c is the number of coarse element faces denoted by Λ_a. The two partitions, $\mathcal{T}_h$ and $\mathcal{T}_c$, are nested. Fig. 4.1 shows a fine grid (finer lines) and a corresponding coarse grid (heavier lines). The log-permeability on the k-th coarse element is a random vector denoted by

$$\boldsymbol{a}_k \equiv \{a_j(\omega) | e_j \subset E_k \}, \tag{4.5}$$

which is referred to as local features and $\boldsymbol{a}_k \subset \boldsymbol{a}$.

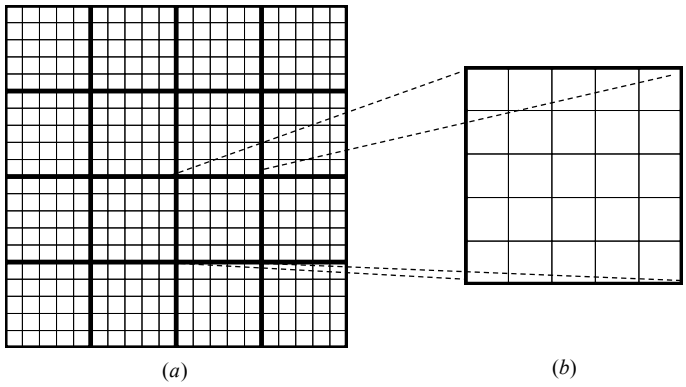

Figure 4.1: Schematic of the domain partition: (a) fine- and coarse-scale grids and (b) fine-scale local region in one coarse element.

4.2 Probabilistic model of responses

A main objective of multiscale modeling is assessing coarse-scale model responses. However, conventional methods such as collocation methods suffer from the curse of dimensionality when the stochastic input $\boldsymbol{a}$ is in a high-dimensional space. In order to overcome this difficulty, a probabilistic graphical model representation of multiscale SPDE is directly constructed in this paper and probability theory is applied to this model to make predictions of model responses.

Without loss of generality, let us suppose that we are interested in model responses on a set of regularly distributed points in the coarse-grid. The coarse-scale model responses associated with these points are denoted by a vector $\boldsymbol{Y} = (y_1, \cdots, y_n)^T$ where y_i stands for model responses. In uncertainty quantification, we aim at estimating the probability distribution of $\boldsymbol{Y}$, i.e. $p(\boldsymbol{Y})$. In a probabilistic framework, this multivariate joint distribution is given by

$$p(\boldsymbol{Y}) = \int p(\boldsymbol{Y}, \boldsymbol{a}) d\boldsymbol{a} = \int p(\boldsymbol{Y}|\boldsymbol{a}) p(\boldsymbol{a}) d\boldsymbol{a}, \tag{4.6}$$

which is based on the knowledge of the stochastic input model and the probabilistic dependence between output and input. In theory, the mapping from $\boldsymbol{a}$ to $\boldsymbol{Y}$ is deterministic given a forward model, which implies that $p(\boldsymbol{Y}|\boldsymbol{a})$ degenerates to delta functions. However, from the perspective of Bayesian statistics, we learn the relationship between input and output completely from training data $\mathcal{D} = \{\boldsymbol{a}^{(t)}, \boldsymbol{Y}^{(t)}\}$. As a result, many other sources of uncertainties, such as the lack of knowledge on forward models and various modeling errors, come into the picture. Then Eq. (4.6) can be approximated as

$$p(\boldsymbol{Y}|\mathcal{D}) = \int p(\boldsymbol{Y}|\boldsymbol{a}, \mathcal{D}) p(\boldsymbol{a}) d\boldsymbol{a}. \tag{4.7}$$

Actually, the training data affect the estimation of the relationship between $\boldsymbol{a}$ and $\boldsymbol{Y}$ through unknown model parameters, $\Theta^\star$, i.e. $p(\boldsymbol{Y}|\boldsymbol{a},\mathcal{D}) \equiv p(\boldsymbol{Y}|\boldsymbol{a},\Theta^\star(\mathcal{D}))$. For notational convenience, we will ignore $\Theta^\star$ temporarily. But it will be brought into the framework later in Section 4.2.2 and will be learned from training data in Section 4.3.

In many practical applications, a stochastic input model can be learned from observation data and thus $p(\boldsymbol{a})$ is assumed to be known. However, $\boldsymbol{a}$ is generally in a high-dimensional space. Although various model reduction techniques might map $\boldsymbol{a}$ to a lower-dimensional space, it is still challenging to construct an accurate probabilistic model of $p(\boldsymbol{Y}|\boldsymbol{a})$ using conventional methods such as kernel density estimation. Therefore, we consider a conditional random field (CRF) representing $p(\boldsymbol{Y}|\boldsymbol{a})$ with a Gibbs distribution as:

$$p(\boldsymbol{Y}|\boldsymbol{a}) \propto \exp\big(-\mathcal{E}(\boldsymbol{Y};\boldsymbol{a})\big), \tag{4.8}$$

where $\mathcal{E}(\boldsymbol{Y};\boldsymbol{a})$ is an 'energy' function. Let I denote the index set of elements in $\boldsymbol{Y}$. Then, the energy function can be written in the following general form

$$\mathcal{E}(\boldsymbol{Y};\boldsymbol{a}) \approx \sum_{i\in I}\phi_i^{(1)}(y_i,\boldsymbol{a}) + \sum_{(i,j)\in I\times I}\phi_{i,j}^{(2)}(y_i,y_j,\boldsymbol{a}) + \cdots + \phi_I^{(|I|)}(\boldsymbol{Y},\boldsymbol{a}),$$

where $\phi^{(n)}$ are feature functions of n variables in $\boldsymbol{Y}$. In this work, we approximate the energy function by ignoring high-order interactions among the model responses. Hence, only $\phi^{(1)}$ and $\phi^{(2)}$ are retained:

$$\mathcal{E}(\boldsymbol{Y};\boldsymbol{a}) \approx \sum_{i\in I}\phi_i(y_i,\boldsymbol{a}) + \sum_{(i,j)\in I\times I}\phi_{i,j}(y_i,y_j,\boldsymbol{a}). \tag{4.9}$$

For notational convenience, the superscripts in potential functions are omitted. Since only up to pairwise interactions between variables in $\boldsymbol{Y}$ are considered in this framework, we further assume that the conditional random field $p(\boldsymbol{Y}|\boldsymbol{a})$ is

a Gaussian Markov random field which can be formulated by [86, 75]

$$p(\boldsymbol{Y}|\boldsymbol{a}) \propto \exp\big(-\sum_i f_i(\boldsymbol{a})y_i - \sum_i \sum_j f_{ij}(\boldsymbol{a})y_i y_j\big), \tag{4.10}$$

where $f_i(\cdot)$ and $f_{ij}(\cdot)$ are functions of the stochastic input.

So far, a global approximation of $p(\boldsymbol{Y}|\boldsymbol{a})$ has been proposed. However, this is of little practical use. Due to the high-dimensional stochastic input, it is difficult to estimate the functions $f_i(\boldsymbol{a})$ and $f_{ij}(\boldsymbol{a})$. Moreover, it is impractical to make predictions directly from this probabilistic model of $\boldsymbol{Y}$. Therefore, the global problem is decomposed into lower-dimensional local sub-problems. Then the approximation of conditional distribution proposed in Eq. (4.10) is applied on each sub-problem. A graphical model associated with the probabilistic model is also presented for making predictions of model responses.

4.2.1 Brief introduction to probabilistic graphical models

Commonly used graphical models include Bayesian networks, undirected graphical models and factor graphs. They define the dependence relationships between random variables in different ways. The main advantage of a graphical model is its ability to factorize the joint distribution over all of the random variables into a product of factors each depending only on a subset of the variables according to the structure of the underlying graph. In this chapter, we focus on undirected graphical models which have undirected links. In an undirected graph, a clique is defined as a subset of nodes such that there exists a link between any pair of nodes in the subset. A maximal clique is a clique that would cease to be a clique when any other nodes are included. Suppose the graphical model is for random variables $\boldsymbol{x}$, i.e. $V = \{x_1, \cdots, x_n\}$. Then, the joint

distribution is factorized as a product of potential functions μ_C over maximal cliques

$$p(\boldsymbol{x}) \propto \prod_C \mu_C(\boldsymbol{x}_C),$$

where C denotes a maximal clique and $\boldsymbol{x}_C$ the variables in that clique.

An undirected graphical model can be converted to a factor graph which provides an efficient way of describing the factorization properties of large graphs [86]. The factor graph, denoted by $G = (V, E, F)$, decomposes a joint probability explicitly by introducing additional factor nodes (functions) F. The joint probability is assumed to be proportional to the product of all factor nodes in the graph. For example, consider the factorization

$$p(\boldsymbol{x}) \propto \prod_{j=1}^{m} \mu_j(\boldsymbol{x}_j),$$

where $\boldsymbol{x}_j \subset \boldsymbol{x}$. In the factor graph for this joint distribution, the factor nodes $F = \{\mu_1, \cdots, \mu_m\}$. A factor node, $\mu_i(\boldsymbol{x}_i)$, is linked to every element in $\boldsymbol{x}_i$. In a typical factor graph, the vertices indicate random variables (or vectors) and squares indicate the factor nodes (see Fig. 4.3b). The factor nodes contain more detailed information about underlying factorization of cliques in an equivalent undirected graphical model. Based on prior knowledge of interacting variables, the factor nodes are properly defined to enable efficient computation of marginal distributions over the joint distribution. This feature will be discussed in the inference problem in Section 4.4. For more details on undirected graphical models, factor graphs and inference on graphical models, the reader can refer to [86, 9, 46].

4.2.2 Probabilistic graphical model for multiscale SPDEs

In this section, a probabilistic model for $p(\boldsymbol{Y}|\boldsymbol{a})$ is constructed with the help of a graphical model. In a multiscale system with stochastic input, the corresponding graph includes nodes for two types of random variables, stochastic input $\boldsymbol{a}$ and output (model responses) $\boldsymbol{Y}$. As discussed in Section 4.1, the stochastic input $\boldsymbol{a}$ is equivalent to $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_{N_c}\}$ in which $\boldsymbol{a}_k$ denotes the local fine-scale features on coarse element E_k and N_c the number of coarse elements. In fluid flow through porous media, the pressure plays an important role in Darcy's law. As in the mixed finite element method, we assume a constant pressure, h_k, on each coarse element. The flow in a coarse element interacts with the flow at the neighboring elements through the adjoint edges. Thus the flux on middle points of edges of coarse elements is also of interest. Finally, the target model responses include the pressure on each coarse element and the fluxes on the middle points of the edges of the coarse elements, i.e. $\boldsymbol{Y} = \{\boldsymbol{u}, \boldsymbol{h}\}$ where $\boldsymbol{h} = \{h_1, \cdots, h_{N_c}\}$ denote the pressure variables and $\boldsymbol{u} = \{u_1, \cdots, u_{M_c}\}$ the fluxes. The spatial distribution of nodes for output and local input features in a graphical model is depicted in Fig. 4.2.

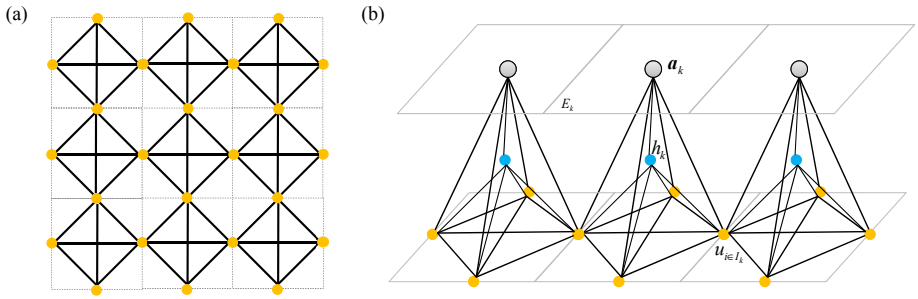

Figure 4.2: (a) Graphical representation of the relationships between model responses, (b) undirected graph for the stochastic input $\boldsymbol{a}$ and model responses $\boldsymbol{Y}$.

As there is little prior information on the relationship between the random

variables $\{\boldsymbol{u}, \boldsymbol{h}\}$ and $\boldsymbol{a}$, all nodes in each coarse element are mutually linked (as shown in Fig. 4.2b). Note that the random vector $\boldsymbol{a}_k$ is treated as a single node in the graph. In order to simplify the graph structure, the spatial correlations between the model responses are taken into account. Each response is only correlated to the response of its neighboring nodes in the same coarse element and long distance interactions are ignored. In this way, the variables on a coarse element, $\{u_{i\in I_k}, h_k, \boldsymbol{a}_k\}$, form a maximal clique in an undirected graphical model. Here, I_k is the index set of responses in element E_k. By the Hammersley-Clifford theorem [4, 48],

$$p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{a}) \propto \prod_k q_k(u_{I_k}, h_k; \boldsymbol{a}_k), \tag{4.11}$$

where $q_k(\cdot)$ is the potential function of the maximal clique on E_k. Thus the global joint distribution is factorized into local potential functions on coarse elements. From the conditional random field defined in Eq. (4.8), $p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{a})$ can be reformulated as

$$p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{a}) \propto \exp\big(-\sum_k \mathcal{E}_k(u_{I_k}, h_k; \boldsymbol{a}_k)\big), \tag{4.12}$$

such that $q_k(u_{I_k}, h_k; \boldsymbol{a}_k) := \exp\big(-\mathcal{E}_k(u_{I_k}, h_k; \boldsymbol{a}_k)\big)$. Then Eq. (4.9) is applied to approximate these local energy functions

$$\begin{aligned} \mathcal{E}_k(u_{I_k}, h_k; \boldsymbol{a}_k) \approx \ & \textstyle\sum_{i\in I_k} \ \phi_{k,i}(u_i, \boldsymbol{a}_k) + \sum_{(i,j)\in I_k\times I_k, i\neq j} \phi_{k,ij}(u_i, u_j, \boldsymbol{a}_k) \\ & + \quad \phi_{k,0}(h_k, \boldsymbol{a}_k) + \sum_{i\in I_k} \phi_{k,i0}(u_i, h_k, \boldsymbol{a}_k), \end{aligned} \tag{4.13}$$

where from the Gaussian Markov field approximation in Eq. (4.10):

$$\begin{aligned} \phi_{k,i}(u_i, \boldsymbol{a}_k) &:= f_{k,i}(\boldsymbol{a}_k)u_i + f_{k,ii}(\boldsymbol{a}_k)u_i^2, \\ \phi_{k,ij}(u_i, u_j, \boldsymbol{a}_k) &:= f_{k,ij}(\boldsymbol{a}_k)u_i u_j, \ i \neq j \\ \phi_{k,0}(h_k, \boldsymbol{a}_k) &:= f_{k,0}(\boldsymbol{a}_k)h_k + f_{k,00}(\boldsymbol{a}_k)h_k^2, \\ \phi_{k,i0}(u_i, h_k, \boldsymbol{a}_k) &:= f_{k,i0}(\boldsymbol{a}_k)u_i h_k. \end{aligned} \tag{4.14}$$

Note that the potential $\phi_{k,ij}$ with $i = j$ is accounted into $\phi_{k,i}$. The functions, $f(\cdot)$, measure the influence of local features on model responses and for stationary permeability are assumed identical on different coarse elements. Nonparametric models are adopted for all functions such that

$$f_{k,\cdot}(\boldsymbol{a}_k) \equiv f_{k,\cdot}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) = \theta_{k,\cdot}^{(1)} + \sum_{t=2}^{r} \theta_{k,\cdot}^{(t)} \zeta_t(\boldsymbol{a}_k), \tag{4.15}$$

where we choose unnormalized Gaussian kernels $\zeta_t(\boldsymbol{a}_k) = \exp(-\frac{\|\boldsymbol{a}_k - \bar{\boldsymbol{a}}_t\|^2}{\sigma_\zeta^2})$ (chapter 6 in [63]). The hyperparameters $\boldsymbol{\theta}_k = \{\theta_{k,\cdot}^{(t)}\}$ are fixed and will be learnt from training data. We also use $\Theta = \bigcup_k \boldsymbol{\theta}_k$ to denote all hyperparameters in the probabilistic model.

Given a set of samples of input $\{\boldsymbol{a}_k^{(n)}\}_{n=1}^N$ and specifying the number of kernels r, the centers of Gaussian kernels, $\{\bar{\boldsymbol{a}}_t\}_{t=1}^r$, are determined using K-means. A typical choice of the kernel width σ_ζ is the average minimum distance between two realizations in the input space, i.e.

$$\sigma_\zeta^2 = \frac{1}{N} \sum_{i=1}^{N} \min_{i \neq j} \|\boldsymbol{a}_k^{(i)} - \boldsymbol{a}_k^{(j)}\|^2. \tag{4.16}$$

Combining Eqs. (4.12) and (4.13), the conditional distribution of model responses is formulated as

$$\begin{aligned} p(\boldsymbol{u}, \boldsymbol{h} | \boldsymbol{a}, \Theta) \propto \prod_k \Big(& \prod_{i \in I_k} \exp\big(- f_{k,i}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) u_i - f_{k,ii}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) u_i^2 \big) \\ & \cdot \prod_{(i,j) \in I_k \times I_k, i \neq j} \exp\big(- f_{k,ij}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) u_i u_j \big) \\ & \cdot \prod_{i \in I_k} \exp\big(- f_{k,i0}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) u_i h_k \big) \\ & \cdot \exp\big(- f_{k,0}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) h_k - f_{k,00}(\boldsymbol{a}_k; \boldsymbol{\theta}_k) h_k^2 \big) \Big). \end{aligned} \tag{4.17}$$

Any realization of the stochastic input $\boldsymbol{a}$ influences the energy function (Eqs. (4.12) and (4.13)) through the function values $f_{k,\cdot}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$. The function

$f_{k,\cdot}$ is a mapping $f_{k,\cdot} : \boldsymbol{a}_k \to \xi_{k,\cdot}$ from local features $\boldsymbol{a}_k$ to a scalar variable $\xi_{k,\cdot}$. In other words, $f_{k,\cdot}$ projects the high-dimensional input into a low-dimensional space. Since these variables $\xi_{k,\cdot}$ are not directly observable, we call them hidden variables in the probabilistic model. The relationships between hidden variables and local features are as follows: $\xi_{k,i} = f_{k,i}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$, $\xi_{k,ii} = f_{k,ii}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$, $\xi_{k,ij} = f_{k,ij}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$, $\xi_{k,0} = f_{k,0}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$, $\xi_{k,00} = f_{k,00}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$ and $\xi_{k,i0} = f_{k,i0}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$, where the hyperparameters $\boldsymbol{\theta}_k = \{\theta_{k,\cdot}^{(t)}\}$ were introduced in Eq. (4.15). The conditional distribution of $\{\boldsymbol{u}, \boldsymbol{h}\}$ can now be formulated in terms of the hidden variables $\boldsymbol{\xi}$:

$$p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi}) \propto \prod_k \Big(\prod_{i \in I_k} \exp(-\xi_{k,i} u_i - \xi_{k,ii} u_i^2) \cdot \prod_{(i,j) \in I_k \times I_k, i \neq j} \exp(-\xi_{k,ij} u_i u_j) \prod_{i \in I_k} \exp(-\xi_{k,i0} u_i h_k) \cdot \exp(-\xi_{k,0} h_k - \xi_{k,00} h_k^2) \Big). \tag{4.18}$$

According to the definition of the hidden variables, each of them is completely fixed given the corresponding local feature $\boldsymbol{a}_k$ and hyperparameters $\boldsymbol{\theta}_k$. In other words, the hidden variables are conditionally independent on local features, e.g. $p(\xi_{k,i}|\xi_{k,j}, \boldsymbol{a}_k, \boldsymbol{\theta}_k) = p(\xi_{k,i}|\boldsymbol{a}_k, \boldsymbol{\theta}_k)$. Thus, we can write the following:

$$\begin{aligned} p(\boldsymbol{\xi}|\boldsymbol{a}, \Theta) &= \prod_k \Big(p(\xi_{k,0}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) p(\xi_{k,00}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) \cdot \\ &\quad \prod_{i \in I_k} p(\xi_{k,i}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) p(\xi_{k,i0}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) \prod_{(i,j) \in I_k \times I_k} p(\xi_{k,ij}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) \Big). \end{aligned} \tag{4.19}$$

Since there exist deterministic relationships between any $\xi_{k,\cdot}$ and $\boldsymbol{a}_k$, $\xi_{k,\cdot}$ takes value at $f_{k,\cdot}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$ with probability 1 given $\boldsymbol{a}_k$. Then the conditional probability of $\xi_{k,\cdot}$ on $\boldsymbol{a}_k$ can be represented with a delta function as in [13]

$$p(\xi_{k,\cdot}|\boldsymbol{a}_k, \boldsymbol{\theta}_k) = \delta\left(\xi_{k,\cdot} - f_{k,\cdot}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)\right). \tag{4.20}$$

As a result, $p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{a}, \Theta)$ can be expressed with hidden variables $\boldsymbol{\xi}$ included as follows:

$$p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{a}, \Theta) = \int p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi}) p(\boldsymbol{\xi}|\boldsymbol{a}, \Theta) d\boldsymbol{\xi}. \tag{4.21}$$

The hidden variables $\boldsymbol{\xi}$ capture fine-scale effects on a coarse-scale. In other words, the influence of high-dimensional stochastic input on responses is represented by $\boldsymbol{\xi}$. Later on, we will discuss that inference on model responses can be implemented directly on $p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi})$ with the information of $\boldsymbol{\xi}$. Thus without involving the high-dimensional input $\boldsymbol{a}$, the computational cost can be significantly reduced. The graphical model with hidden variables is shown in Fig. 4.3(a).

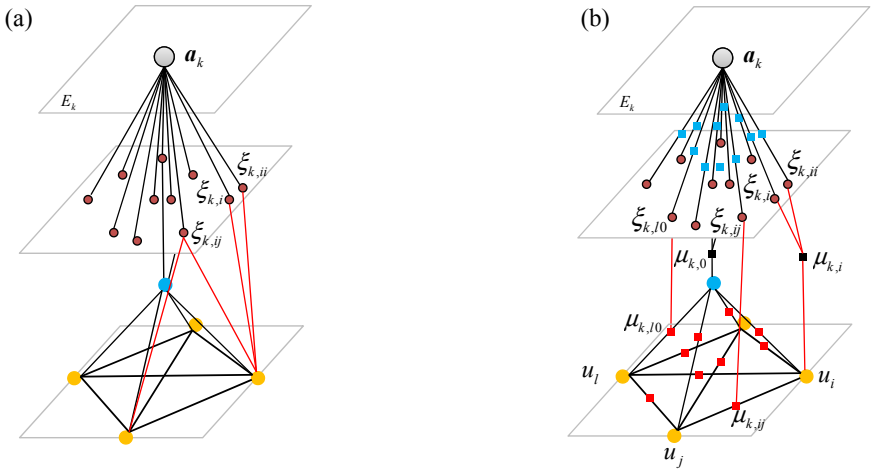

Figure 4.3: (a) Undirected graphical model with hidden variables, (b) an equivalent factor graph.

In order to factorize the joint probability $p(\boldsymbol{u}, \boldsymbol{h}, \boldsymbol{a}|\Theta)$ explicitly via the graphical model, the undirected graph with hidden variables in Fig. 4.3(a) is transformed into a factor graph in Fig. 4.3(b) in which the factor nodes are potential functions defined through the energy functions in Eq. (4.13), i.e.

$$\begin{aligned}
\mu_{k,i} &:= \exp(-\phi_{k,i}(u_i, \xi_{k,i}, \xi_{k,ii})), \\
\mu_{k,ij} &:= \exp(-\phi_{k,ij}(u_i, u_j, \xi_{k,ij})), \\
\mu_{k,0} &:= \exp(-\phi_{k,0}(h_k, \xi_{k,0}, \xi_{k,00})), \\
\mu_{k,i0} &:= \exp(-\phi_{k,i0}(u_i, h_k, \xi_{k,i0})).
\end{aligned} \tag{4.22}$$

The potential functions for any of the hidden variables $\boldsymbol{\xi}$ and local feature $\boldsymbol{a}_k$ are simply the conditional distributions (delta functions) in Eq. (4.20).

Remark 1: The subscripts of feature functions $\phi(\cdot)$ in Eq. (4.13), potential functions $\mu(\cdot)$ in Eq. (4.22), hidden variables ξ as well as coefficient functions $f(\cdot)$ are defined according to the following rules: (k, i) denotes the flux component with flux index i, (k, ij) denotes interaction terms between the fluxes u_i and u_j ($i \neq j$), $(k, 0)$ denotes the pressure h_k on the coarse element k and $(k, i0)$ denotes interaction terms between the flux u_i and pressure h_k. All these quantities are defined on the k-th coarse element. Note that the index set for flux, I, is a set of positive integers such that $I = \{1, 2, \cdots, M_c\}$ where $M_c = \dim(\boldsymbol{u})$ is the number of model responses for flux. We use 0 to indicate quantities related to pressure.

Remark 2: The hidden variables $\boldsymbol{\xi}$ on different coarse elements k are different. For a stationary permeability random field $\boldsymbol{a}$, one can assume that the hyperparameters $\boldsymbol{\theta}_k$ are the same on different coarse elements. This is because local features $\boldsymbol{a}_k$ have the same distribution on coarse elements and one should expect the same relationship between hidden variables and local input on different elements. Thus one can assume that the hidden variables on different elements have the same marginal distribution. However, they cannot be treated as the same variables, $\boldsymbol{\xi}_1 = \boldsymbol{\xi}_2 = \cdots = \boldsymbol{\xi}_{N_c}$, as they are associated with local features $\boldsymbol{a}_k$. Given a realization of stationary stochastic input $\boldsymbol{a}^{(i)}$, the local features $\boldsymbol{a}_k^{(i)}$ and $\boldsymbol{a}_l^{(i)}$ on elements E_k and E_l are generally different. Consider the realizations of two hidden variables, $\xi_{k,ij}$ and $\xi_{l,pq}$ such that $\xi_{k,ij} = f_{k,ij}(\boldsymbol{a}_k; \boldsymbol{\theta}_k)$ and $\xi_{l,pq} = f_{l,pq}(\boldsymbol{a}_l; \boldsymbol{\theta}_l)$. Even though the hyperparameters $\boldsymbol{\theta}_k = \boldsymbol{\theta}_l$, the realizations of the hidden variables are generally different. If we have a nonstationary random field, the hidden variables on different elements are different variables with dif-

ferent marginal distributions.

4.3 Graphical model parameter learning

We proceed next to learn the various parameters that define the probabilistic graphical model. Suppose we have a training set $\mathcal{D} = \{\boldsymbol{a}^{(t)}, \boldsymbol{u}^{(t)}, \boldsymbol{h}^{(t)}\}_{t=1}^{N}$ where $\boldsymbol{a}^{(t)} = \{\boldsymbol{a}_k^{(t)}\}_{k=1}^{N_c}$. The likelihood function of training data is formulated in Eq. (4.12), which can be also written as

$$\begin{aligned} p(\mathcal{D}|\Theta) &= \prod_{t=1}^{N} p(\boldsymbol{u}^{(t)}, \boldsymbol{h}^{(t)}|\boldsymbol{a}^{(t)}, \Theta) p(\boldsymbol{a}^{(t)}) \\ &\propto \prod_{t=1}^{N} \exp\Big(-\sum_{k} \mathcal{E}_k(u_{I_k}^{(t)}, h_k^{(t)}; \boldsymbol{a}_k^{(t)}, \boldsymbol{\theta}_k)\Big) \\ &\propto \prod_{k} \exp\Big(-\sum_{t=1}^{N} \mathcal{E}_k(u_{I_k}^{(t)}, h_k^{(t)}; \boldsymbol{a}_k^{(t)}, \boldsymbol{\theta}_k)\Big), \end{aligned} \tag{4.23}$$

where $\boldsymbol{\theta}_k$ are constant hyperparameters in element E_k (defined in Eq. (4.15)). By specifying the prior $p(\Theta)$, the Bayesian posterior of hyperparameters in the probabilistic model is

$$p(\Theta|\mathcal{D}) \propto p(\mathcal{D}|\Theta) p(\Theta). \tag{4.24}$$

In this work, we set the prior as a multivariate Gaussian distribution with mean zero and an identity covariance matrix. Thus the elements in Θ are mutually independent in the prior, which leads to $p(\Theta) = \prod_k p(\boldsymbol{\theta}_k)$. The posterior distribution in Eq. (4.24) can then be decomposed as follows:

$$\begin{aligned} p(\Theta|\mathcal{D}) &= \prod_{k} p(\boldsymbol{\theta}_k|\mathcal{D}) \\ &\propto \prod_{k} \exp\Big(-\sum_{t=1}^{N} \mathcal{E}_k(u_{I_k}^{(t)}, h_k^{(t)}; \boldsymbol{a}_k^{(t)}, \boldsymbol{\theta}_k)\Big) p(\boldsymbol{\theta}_k). \end{aligned} \tag{4.25}$$

As a result, Θ in the probabilistic graphical model can also be estimated locally. We will use a special Monte Carlo method – Sequential Monte Carlo (SMC) [61] to estimate the parameters $\boldsymbol{\theta}_k$ on each coarse element through the local posterior distribution $p(\boldsymbol{\theta}_k|\mathcal{D}_k)$, where $\mathcal{D}_k$ are training data related to model responses on element E_k. The parameter values, $\Theta^\star$, that maximize the posterior will be taken as fixed parameter values in the probabilistic model to perform probabilistic inference with regard to model responses.

Often posterior distributions can be multi-modal. Conventional MCMC will be trapped by the local modes and long mixing times will be required making it inefficient. In order to explore multi-modal posteriors efficiently, a SMC method is employed [61, 51, 6, 39, 87]. First, the idea of annealing/tempering is introduced. Given the target posterior distribution π_n, a sequence of auxiliary distributions, $\{\pi_0, \cdots, \pi_n\}$, is proposed to move smoothly from a tractable distribution π_0 to the target π_n. Let the target distribution be the local posterior $p(\boldsymbol{\theta}_k|\mathcal{D}_k)$ in Eq. (4.25). Then, the following auxiliary distributions are adopted:

$$\pi_t(\boldsymbol{\theta}_k) \propto p^{\gamma_t}(\mathcal{D}_k|\boldsymbol{\theta}_k)p(\boldsymbol{\theta}_k), \tag{4.26}$$

where $t = 0, 1, \cdots, n$ and $0 = \gamma_0 < \gamma_1 < \cdots < \gamma_n = 1$ are tempering parameters and $p(\mathcal{D}_k|\boldsymbol{\theta}_k) \propto \exp\Big(-\sum_{t=1}^{N} \mathcal{E}_k(u_{I_k}^{(t)}, h_k^{(t)}; \boldsymbol{a}_k^{(t)}, \boldsymbol{\theta}_k)\Big)$ according to Eq. (4.25). To simplify the notation, in the following discussion of the SMC algorithm, we use θ to denote the hyperparameters $\boldsymbol{\theta}_k$.

Remark 3: When the stochastic input $\boldsymbol{a}$ is a stationary random field, the local features have the same joint distribution, $p(\boldsymbol{a}_1) = p(\boldsymbol{a}_2) = \cdots = p(\boldsymbol{a}_{N_c})$. In this case, we assume that the hidden variables have the same relationships with the local features in all coarse elements k, i.e. $\Theta \equiv \boldsymbol{\theta}_1 = \boldsymbol{\theta}_2 = \cdots = \boldsymbol{\theta}_{N_c}$. Thus the global posterior $p(\Theta|\mathcal{D})$ in Eq. (4.24) is directly used to infer the hyperparame-

ters. Let $\pi_n \equiv p(\Theta|\mathcal{D})$, then the auxiliary distributions are defined as

$$\pi_t(\Theta) \propto p^{\gamma_t}(\mathcal{D}|\Theta)p(\Theta). \tag{4.27}$$

The SMC method takes samples from such a sequence of probability distributions based on importance sampling and resampling techniques and constructs a sequential Bayesian inference. In this chapter, we follow the algorithms described in section 2.2.2, Chapter 2.

4.4 Inference on probabilistic graphical models

The model proposed in Section 4.2 enables us to perform probabilistic inference with regard to model responses. Suppose the probability distribution of stochastic input $p(\boldsymbol{a})$ is known, we are then interested in the marginal distributions of responses $p(u_i)$ or $p(h_k)$. This task is challenging as direct marginalization over random variables in the joint distribution is generally intractable. MC methods are extensively used but the convergence rate is slow. Variational methods require discovering good approximating functions [101].

In this work, we will address this problem using belief propagation (BP) — an efficient way of computing marginals of probability distributions from a graphical model. The BP algorithms propagate information through a graphical model via messages between neighboring nodes, which is equivalent to applying a local message-passing algorithm [67]. Consider the general factor graph in Fig. 4.4. Denote all the factor nodes directly linked to variable x_i by $\Gamma(x_i)$. At iteration n of the BP algorithm, the message from x_i to factor node μ is a function

of x_i and the update rule is

$$m^{(n)}_{x_i \to \mu}(x_i) \leftarrow \mu_i(x_i) \prod_{\mu_{pi} \in \Gamma(x_i) \backslash \mu} m^{(n-1)}_{\mu_{pi} \to x_i}(x_i). \tag{4.28}$$

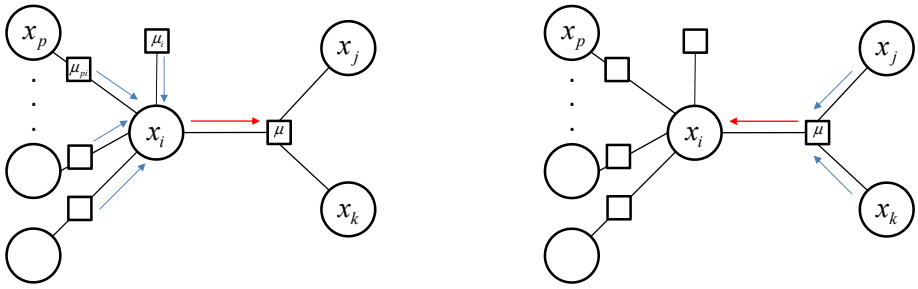

Figure 4.4: Message propagation in a factor graph (a) message passing from a variable node to a factor node, (b) message passing from a factor node to a variable node.

Denote by $\boldsymbol{x}_\mu$ the neighboring variables directly linked to factor node μ, the message from a factor node μ to variable x_i is a function of x_i which is updated by

$$m^{(n)}_{\mu \to x_i}(x_i) \leftarrow \int_{\boldsymbol{x}_\mu \backslash x_i} \mu(\boldsymbol{x}_\mu) \prod_{x_t \in \boldsymbol{x}_\mu \backslash x_i} m^{(n)}_{x_t \to \mu}(x_t) d\boldsymbol{x}_\mu \backslash x_i. \tag{4.29}$$

When the underlying factor graph is a tree, one iteration of BP is guaranteed to compute the correct posterior marginal distribution of x_i [100]. When the factor graph contains loops, the messages must be updated iteratively until convergence is achieved. An estimate of the posterior marginal distribution of x_i at each iteration is obtained by multiplying all incoming messages from neighboring factor nodes:

$$p^{(n)}(x_i) \propto \mu_i(x_i) \prod_{\mu \in \Gamma(x_i)} m^{(n)}_{\mu \to x_i}(x_i). \tag{4.30}$$

Furthermore, this estimation can be extended to joint probability density functions (PDFs). Replacing the variable x_i with a set of correlated nodes $\boldsymbol{x}$, the joint PDF can be estimated by multiplying all incoming messages to each

element variables in $\boldsymbol{x}$ with the factor node $\mu(\boldsymbol{x})$, i.e.

$$p^{(n)}(\boldsymbol{x}) \propto \mu(\boldsymbol{x}) \prod_{x_j \in \boldsymbol{x}} \prod_{\mu \in \Gamma(x_j)} m^{(n)}_{\mu \to x_j}(x_j). \tag{4.31}$$

Applying BP algorithms to the factor graph in Fig. 4.3(b), we can obtain marginal distributions $p(u_i)$ and $p(h_k)$ without generating samples of stochastic input $\boldsymbol{a}$ or calling the deterministic solver. The challenge lies in computing the message from a factor node to a neighboring variable when fine-scale features $\boldsymbol{a}_k$ are included in the factor node. In this case, a high-dimensional integration is required according to Eq. (4.29). In order to solve this problem, the fine-scale features are integrated out of the probabilistic model such that

$$\begin{aligned} p(\boldsymbol{u}, \boldsymbol{h}, \boldsymbol{\xi}) &= \int p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi}) p(\boldsymbol{\xi}|\boldsymbol{a}) p(\boldsymbol{a}) d\boldsymbol{a} \\ &= p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi}) p(\boldsymbol{\xi}). \end{aligned} \tag{4.32}$$

Then the factor graph in Fig. 4.3(b) is transformed to the graph in Fig. 4.5(a). The hidden variables are correlated in $p(\boldsymbol{\xi})$ and thus are connected in the graphical model.

In the BP algorithm used in this work, the messages are updated in parallel. At each iteration, we calculate the messages from each factor node to its neighboring variable nodes as well as the messages from each variable node to its neighboring factor nodes based on messages updated in the previous iteration [62]. The messages are considered as converged if their change is less than a threshold in two successive iterations. In the graphical model in Fig. 4.5(a), there exists a unique message between any factor node (potential function) and any of its arguments, including (1) messages between factor node $\mu_{k,ij}$ and any of its arguments, u_i, u_j or $\xi_{k,ij}$; (2) messages between factor node $\mu_{kk,l}$ and any of its arguments, u_l, h_k or $\xi_{kk,l}$; (3) messages between factor node $\mu_{k,i}$ and any

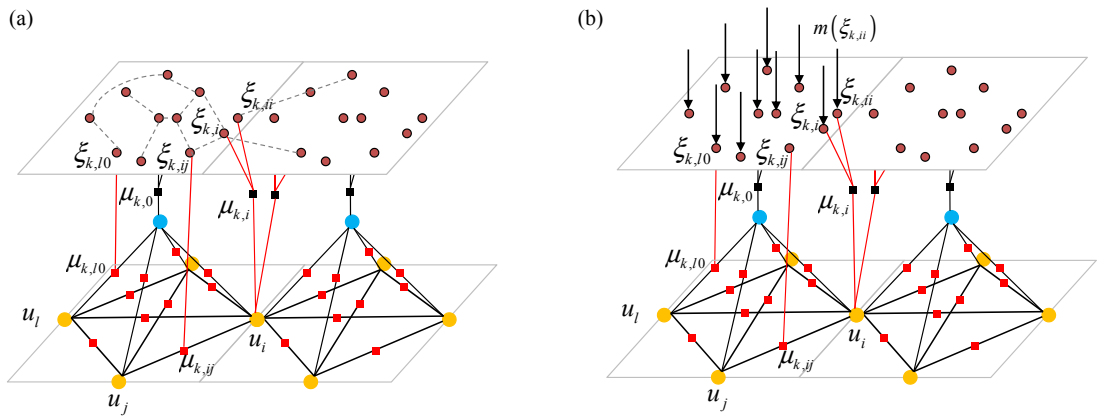

Figure 4.5: (a) Reduced factor graph of the probabilistic graphical model in Fig. 4.3(b) in which the stochastic input $\boldsymbol{a}$ is integrated out, (b) the correlation between hidden variables can be ignored in belief propagation by a direct iterative update of incoming messages $m(\xi_{k,ij})$.

of its arguments, u_i, $\xi_{k,i}$ or $\xi_{k,ii}$; (4) messages between factor node μ_{kk} and any of its arguments, h_k, ξ_{kk} or ξ_{kkk}; (5) messages between hidden variables. The messages in (5) are more complex and will be discussed separately later on. The messages in (1)-(4), since there is no prior information, are represented nonparametrically as weighted Gaussian mixtures. Without loss of generality, consider the message from factor node $\mu_{k,ij}(u_i, u_j, \xi_{k,ij})$ to variable node u_i, which is approximated by

$$m_{\mu_{k,ij}\to u_i}(u_i) \approx \sum_{t=1}^{T} l_t \mathcal{N}(u_i; \bar{u}_i^t, \sigma_i^2), \tag{4.33}$$

where l_t is the weight of a Gaussian kernel with mean $\bar{u}_i^t$ and variance σ_i^2 [83]. At iteration n of the BP algorithm, the messages between factor nodes and variables nodes are updated according to Eqs. (4.28) and (4.29), i.e.

$$m^{(n)}_{\mu_{k,ij}\to u_i}(u_i) \leftarrow \int \mu_{k,ij}(u_i, u_j, \xi_{k,ij}) m^{(n)}_{u_j\to\mu_{k,ij}}(u_j) m^{(n)}_{\xi_{k,ij}\to\mu_{k,ij}}(\xi_{k,ij}) d\xi_{k,ij} du_j, \tag{4.34}$$

and the message $m^{(n)}_{u_i\to\mu_{k,ij}}(u_i)$ is simply the product of all incoming messages

from neighboring factor nodes to u_i at iteration $n-1$. However, as discussed in [83], a BP update which multiplies d Gaussian mixtures, each containing T components can produce a Gaussian mixture with T^d components, i.e. the number of mixture components increases exponentially. Therefore, when updating the message $m_{\mu_{k,ij}\to u_i}(u_i)$ in Eq. (4.34), we draw samples $(u_i, u_j, \xi_{k,ij})$ from $\mu_{k,ij}(u_i, u_j, \xi_{k,ij}) \cdot m^{(n)}_{u_j\to\mu_{k,ij}}(u_j) \cdot m^{(n)}_{\xi_{k,ij}\to\mu_{k,ij}}(\xi_{k,ij})$ using MCMC. Then the message is approximated using a Gaussian mixture model with T kernels as in Eq. (4.33) from samples of u_i [60].

The main remaining challenge lies in the update of messages between the hidden variables $\boldsymbol{\xi}$. Although analytic expressions of $p(\boldsymbol{a})$ and $p(\boldsymbol{\xi}|\boldsymbol{a})$ are explicit, the joint distribution of hidden variables $\boldsymbol{\xi}$ could be complicated such that the links between them are implicit when stochastic input has been removed from the graphical model. To bypass the difficulties in passing messages between hidden variables, we examine the four messages related to a hidden variable $\xi_{k,ij}$, i.e. (1) the message from a factor node $\mu_{k,ij}$, denoted by $m_{\mu_{k,ij}\to\xi_{k,ij}}$, (2) the message from $\xi_{k,ij}$ to factor node $\mu_{k,ij}$, (3) the message from other hidden variables, denoted by $m(\xi_{k,ij})$, and (4) the messages from $\xi_{k,ij}$ to other hidden variables. According to Fig. 4.5(a) and Eq. (4.28), the message in (2) equals to the message in (3). On the other hand, according to Eq. (4.30), the messages in (1) and (3) are correlated in the following way

$$p(\xi_{k,ij}) \propto m(\xi_{k,ij}) m_{\mu_{k,ij}\to\xi_{k,ij}}(\xi_{k,ij}). \tag{4.35}$$

The marginal distributions of hidden variables, $p(\xi_{k,\cdot})$ can be obtained by standard kernel density estimators given samples of $\boldsymbol{a}_k$ (see Eq. (4.15)) and are thus known. As a result, the messages between hidden variables are updated directly

via $p(\xi_{k,ij})$, i.e. at iteration n, the input messages $m(\xi_{k,ij})$ are updated by

$$m^{(n+1)}(\xi_{k,ij}) \propto p(\xi_{k,ij})/m^{(n)}_{\mu_{k,ij}\to\xi_{k,ij}}(\xi_{k,ij}). \tag{4.36}$$

Then the only messages undetermined are those from $\xi_{k,ij}$ to other hidden variables. In theory, they are used to compute the incoming messages to other hidden variables, which, however, can be estimated in the same way as in Eq. (4.36). Therefore, the messages between hidden variables do not play any role in belief propagation and the graphical model in Fig. 4.5(a) is transformed into the one in Fig. 4.5(b). Finally, starting from initial guess of all messages, the BP algorithm iteratively updates the messages until the marginal distributions of responses $\{u_i\}$ and $\{h_k\}$ converge. The procedure is summarized in Algorithm 1.

Algorithm 1: *A general Belief Propagation with nonparametric messages*

1. Initialization: Set initial input messages of hidden variables as their marginal distributions obtained from sampling, i.e. $m^0(\xi_{k,\cdot}) = p(\xi_{k,\cdot})$ and all the other messages as Gaussian mixtures defined in Eq. (4.33).

2. Iterate: At step n, update messages according to Eqs. (4.34) and (4.36).

3. Convergence: The marginal distributions $p(u_i)$ (and also for $p(h_k)$) are approximated by

$$p(u_i) \propto \prod_{\mu\in\Gamma(u_i)} m_{\mu\to i}(u_i) \approx \sum_{t=1}^{T} l_t \mathcal{N}(u_i; \bar{u}_i^t, \sigma_i^2),$$

from the sampling-based method. The marginal distributions $\{p(u_i)\}$ converge when $\max \|\bar{u}_i^t - \bar{u}_i^{t-1}\| < \epsilon$. Stop iteration when all marginal distributions converge.

Remark 4: Given a realization of the stochastic input, $\boldsymbol{a}^{(n)}$, the values of the hidden variables, e.g. $\xi_{k,ij}^{(n)}$, can be directly obtained through the function $f_{k,ij}$,

i.e. $\xi_{k,ij}^{(n)} = f_{k,ij}(\boldsymbol{a}_k^{(n)})$. As the hidden variables in Fig. 4.5 are observed, there is no message $m(\xi_{\cdot,\cdot})$ between them. Then the factor graph in Fig. 4.5(b) corresponds to the conditional distribution $p(\boldsymbol{u}, \boldsymbol{h}|\boldsymbol{\xi}^{(n)})$. The unobserved variables in this graph are the model responses $(\boldsymbol{u}, \boldsymbol{h})$. When belief propagation is performed, we obtain the marginal distributions of the model responses conditioned on the input, e.g. $p(u_i|\boldsymbol{a}^{(n)})$ and $p(h_k|\boldsymbol{a}^{(n)})$. As a result, let the expectations $\mathbb{E}(u_i|\boldsymbol{a}^{(n)})$ and $\mathbb{E}(h_k|\boldsymbol{a}^{(n)})$ be the predicted values of model responses. We can then obtain a surrogate model by running the belief propagation algorithm on a factor graph given a realization of the stochastic input.

4.5 Numerical examples

In this section, we construct probabilistic graphical model based solutions to predict fluid flow in random heterogeneous porous media. The domain is a unit square $[0, 1]^2$. The permeability is defined on a 64×64 fine grid and we are interested in the flux at the middle point of edges of coarse elements as well as the pressure on a 8×8 coarse grid. The model responses in the training data, $\mathcal{D} = \{\boldsymbol{a}^{(i)}, \boldsymbol{h}^{(i)}, \boldsymbol{u}^{(i)}\}$, are generated using a mixed finite element method on the fine grid and are collected on locations related to the coarse grid [70, 3]. We choose $r = 4$ kernels in Eq. (4.15) to approximate the relationship between the hidden variables and the local features. Since there are 20 hidden variables on each coarse element, there are totally 80 hyperparameters in $\boldsymbol{\theta}_k$ associated with each coarse element. In SMC learning of these hyperparameters, we choose standard Gaussian distribution $\mathcal{N}(0, 1)$ as the prior for each component of hyperparameters. The threshold of ESS is set to be $\text{ESS}_{\min} = 0.85N$. A linear

cooling schedule is selected for γ_t in Eq. (4.27). For 500 time steps, the sequence $\{\gamma_0, \cdots, \gamma_{500}\}$ increases uniformly from 0 to 1. In SMC, we employ 800 particles.

4.5.1 Isotropic random field

In this example, the log-permeability $\boldsymbol{a}$ is a Gaussian random field with mean zero and an exponential covariance function defined as

$$\mathrm{cov}(\boldsymbol{x}, \boldsymbol{x}^\star) = \sigma^2 \exp(-\frac{|x_1 - x_1^\star|}{L_1} - \frac{|x_2 - x_2^\star|}{L_2}), \tag{4.37}$$

where coordinates $\boldsymbol{x} = (x_1, x_2)$ and $\boldsymbol{x}^\star = (x_1^\star, x_2^\star)$ and σ is the standard deviation of the random field. An isotropic random field is assumed such that correlation lengths $L_1 = L_2 = 0.1$ and the standard deviation $\sigma = 1.0$. The samples of permeability are generated using standard Karhunen-Loève (KL) expansion with the first 100 terms. Since this is a stationary random field, the local features $\boldsymbol{a}_k$ are subject to the same distribution. As hidden variables capture local features on each coarse element, it is reasonable to assume the same relationships between the hidden variables and the local features on different coarse elements. Thus the hyperparameters defined in Eq. (4.15) on coarse elements are identical, i.e. $\boldsymbol{\theta} = \boldsymbol{\theta}_1 = \cdots = \boldsymbol{\theta}_{N_c}$ and $\Theta \equiv \boldsymbol{\theta}$ in Eq. (4.23). The global posterior distribution, $p(\Theta|\mathcal{D})$, defined in Eq. (4.23) is directly used to estimate the hyperparameters Θ. In this example, $N = 20,\ 40,\ 60$ training data are generated to train the probabilistic model.

The source term f in Eq. (4.3) is used to model injection/production wells:

$$f(\boldsymbol{x}) = \begin{cases} -r, & \text{if } 0 \leq x_i < w, \text{ for } i = 1,2, \\ r, & \text{if } 1 - w \leq x_i < 1, \text{ for } i = 1,2, \\ 0 & \text{otherwise.} \end{cases} \tag{4.38}$$

The parameters are chosen to be $r = 10$ and $w = 1/8$. No-flow homogeneous Neumann boundary conditions are applied on all boundaries. The threshold for BP convergence is set to $\epsilon = 10^{-4}$. Note that the reference solutions (mean, variance and marginal PDFs) are obtained by MC simulation with 10^6 samples.

According to the belief propagation algorithm proposed in Section 4.4, the estimated marginal distributions are Gaussian mixtures (or Gaussian if there is only one Gaussian component in each message). In order to verify the correctness of probabilistic graphical models, we randomly generate a realization of stochastic input and predict the model responses using belief propagation as discussed in Remark 4. The messages in belief propagation are assumed to be Gaussian functions. The graphical model is trained with 60 data points. In comparison with the direct simulation results obtained from the mixed multiscale FEM (Fig. 4.6), we can see that accurate predictions are obtained from the probabilistic graphical model given an observation of stochastic input. To quantitatively estimate the predictive accuracy of the probabilistic graphical model, a k-fold ($k = 10$) cross-validation is applied [43]. It is performed with 40 and 60 samples, respectively (Fig. 4.7). On each fold, the mean squared prediction error for each element of model responses is obtained and the average of errors is taken on the k folds.

Next, we use belief propagation to estimate the statistics of the model responses without given realizations of input, i.e. all variables in Fig. 4.5 are unobserved. Four Gaussian mixture components are adopted in the messages in belief propagation. The predicted mean and variance of the model responses (velocities and pressure) are compared with the MC solutions in Figs. 4.8 to 4.13. The comparison shows that more training data can generate probabilistic mod-

els with higher predictive accuracy. The convergence plot in Fig. 4.14 shows that the error in the variance of the flux predicted by the probabilistic graphical model decreases much faster than that in the standard MC simulation. The reference solution here is taken as Monte Carlo with 10^6 samples. With a stationary random field as stochastic input, the hyperparameters do not vary on coarse elements. As a result, they can be accurately estimated with a small number of data sets. However, the same number of samples is far from sufficient for convergence in the MC simulation.

In the belief propagation algorithm, the number of Gaussian components in the nonparametric messages should be specified. In this example, we set the number of components as 2 and 4 and predict the marginal distributions for each case. Fig. 4.15 shows the predicted PDFs of the x-velocity at point $(0.5, 0.4375)$ in the spatial domain. Fig. 4.16 shows the predicted PDFs of y-velocity at point $(0.4375, 0.5)$ and Fig. 4.17 shows the predicted PDFs of pressure on a coarse element centered at $(0.4375, 0.4375)$. Obviously, when the target marginal PDF obtained from MC simulation is non-Gaussian, the assumption of Gaussian messages cannot apply. With sufficient training data, belief propagation with 4 components in the messages generally gives better prediction of the marginal PDF than that with only 2 components in the messages. In addition to the first-order marginal PDFs, the probabilistic graphical model can also capture the correlations of model responses. Replacing the variable in Eq. (4.30) with a pair of connected nodes, the joint PDF can be estimated by multiplying all incoming messages to both nodes with the factor node between them. In Fig. 4.18, the joint distributions of x-velocity and y-velocity at two different locations are presented. Compared with results from direct simulations, the non-Gaussian joint distributions are accurately estimated with a probabilistic graphical model

trained by 60 data points.

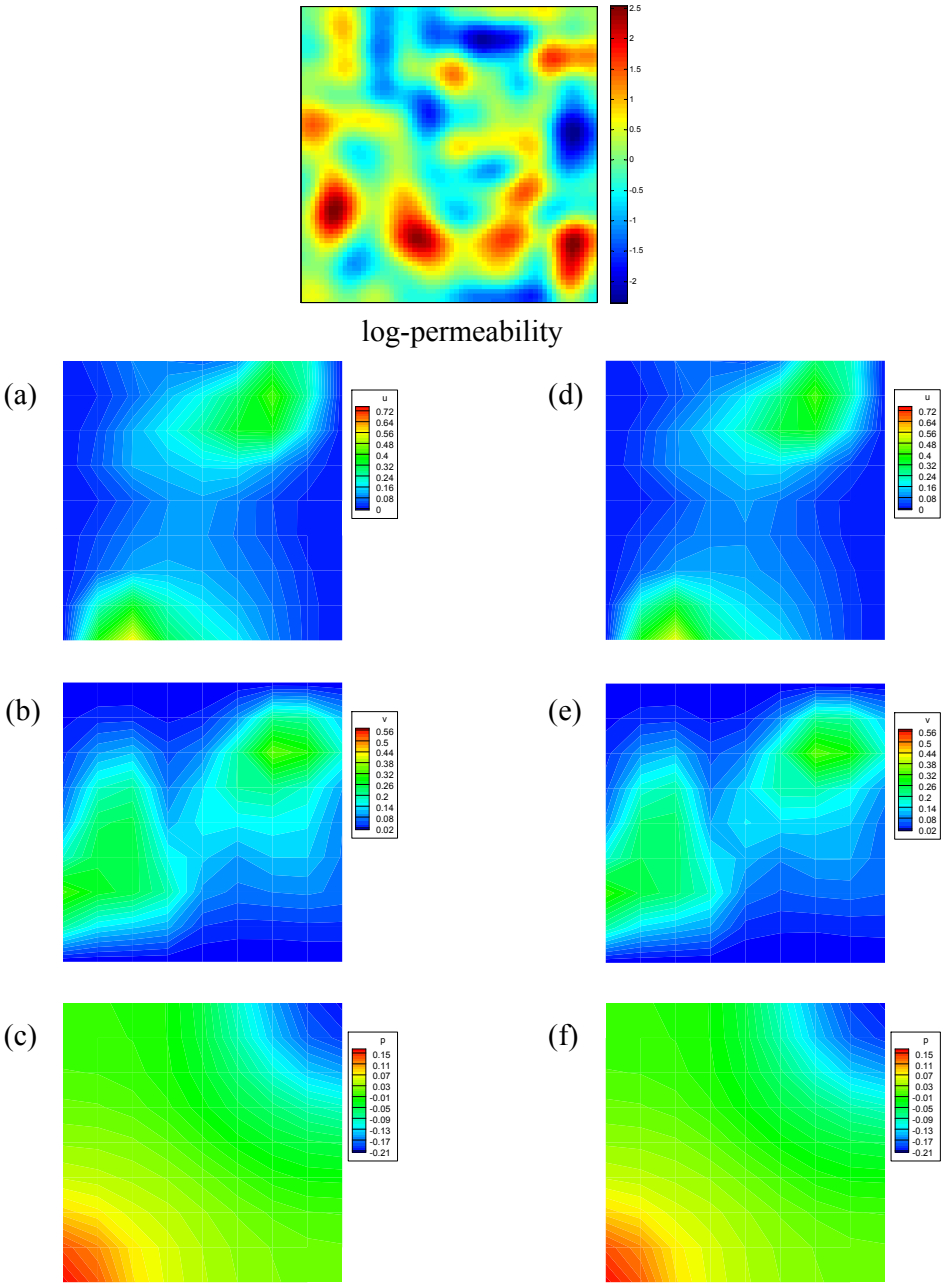

Figure 4.6: Isotropic Random Field: Predicted values of model responses given a realization of the stochastic input (a)-(c) x-velocity, y-velocity and pressure obtained from the direct simulation, and (d)-(f) x-velocity, y-velocity and pressure predicted by the probabilistic graphical model (trained with 60 data points).

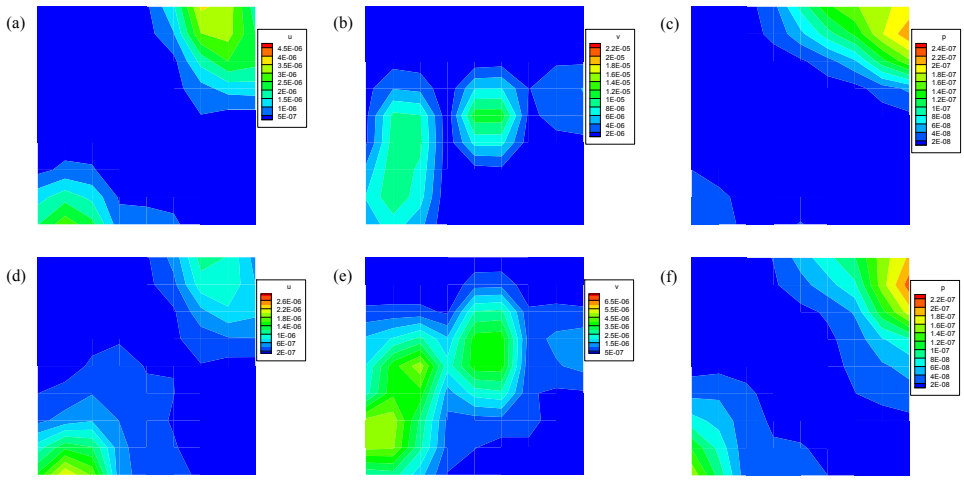

Figure 4.7: Isotropic Random Field: k-fold cross-validation error ($k = 10$) of x-velocity, y-velocity and pressure predicted by the probabilistic graphical model with (a)-(c) 40 samples, and (d)-(f) 60 samples.

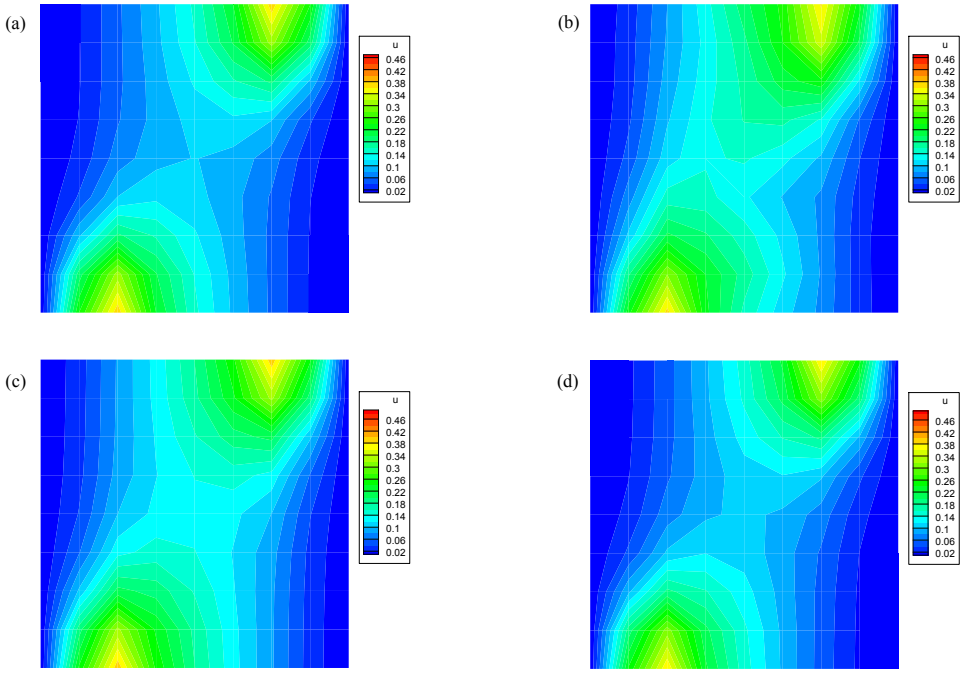

Figure 4.8: Isotropic Random Field: Predicted mean of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

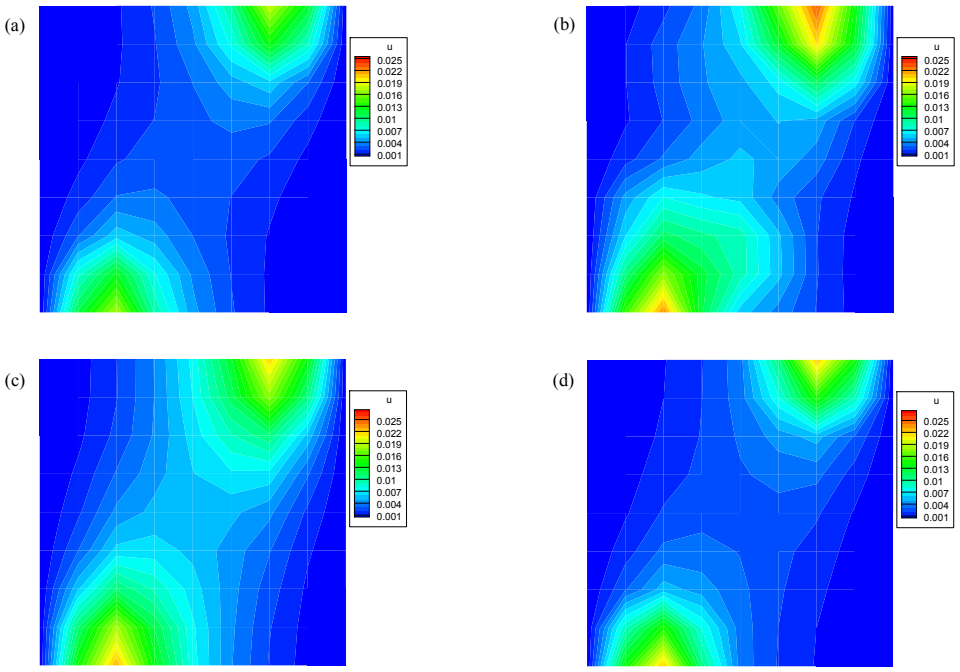

Figure 4.9: Isotropic Random Field: Predicted variance of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

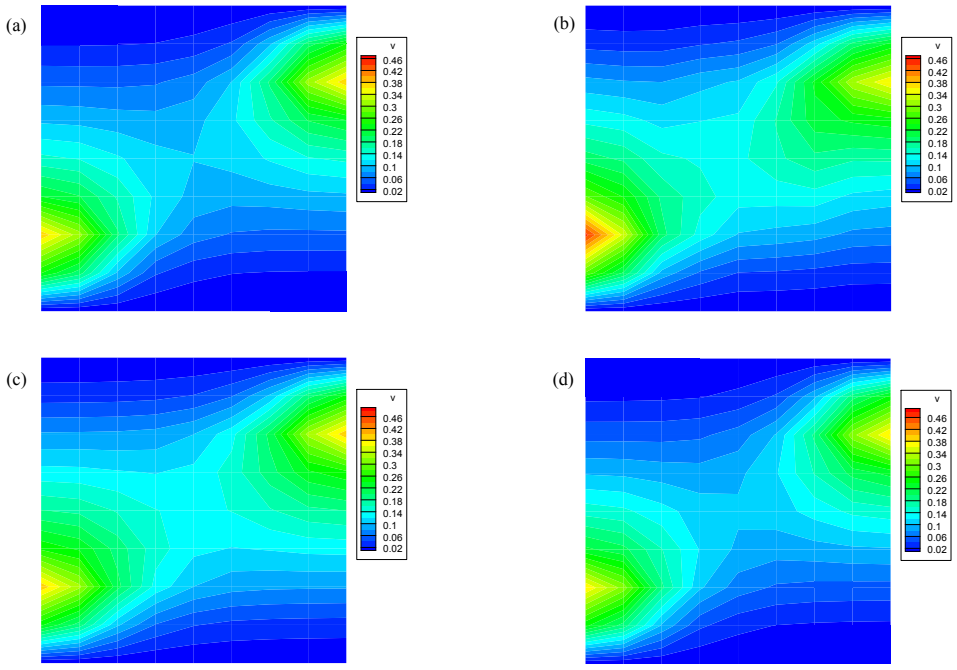

Figure 4.10: Isotropic Random Field: Predicted mean of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

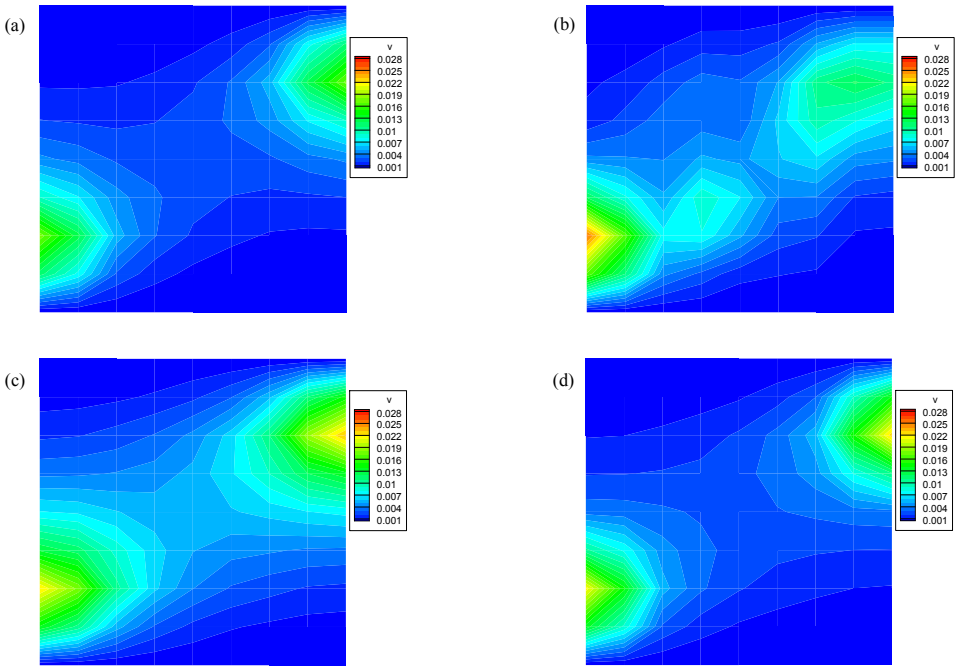

Figure 4.11: Isotropic Random Field: Predicted variance of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

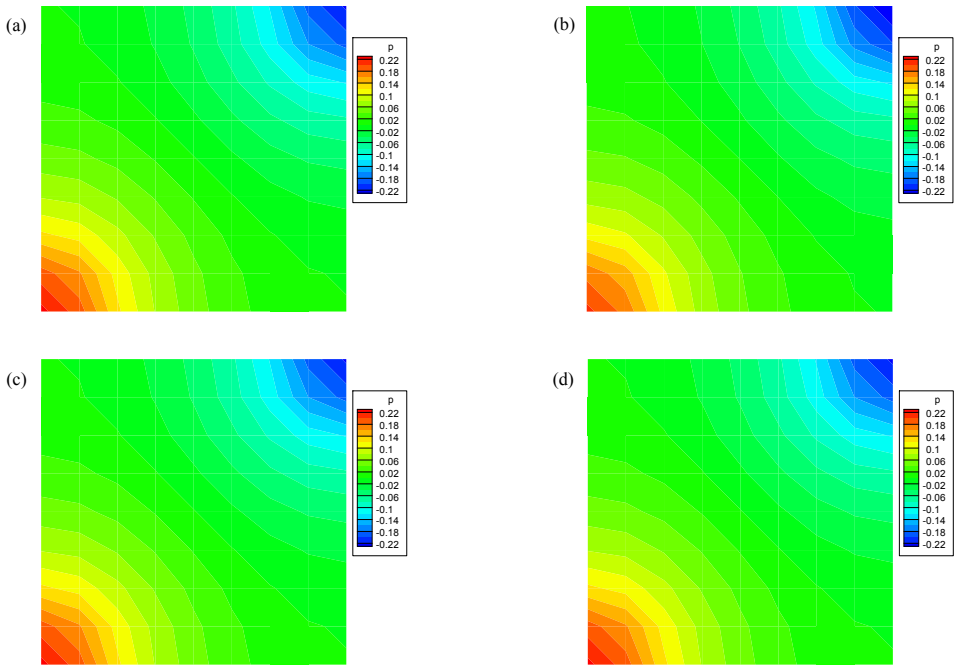

Figure 4.12: Isotropic Random Field: Predicted mean of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

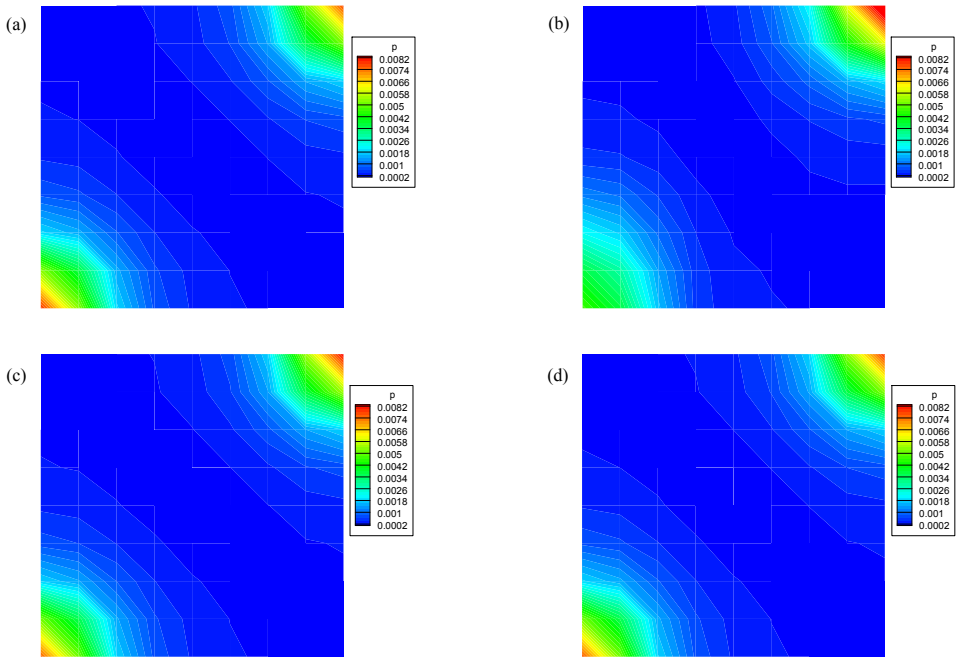

Figure 4.13: Isotropic Random Field: Predicted variance of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 20, (c) 40 and (d) 60 data.

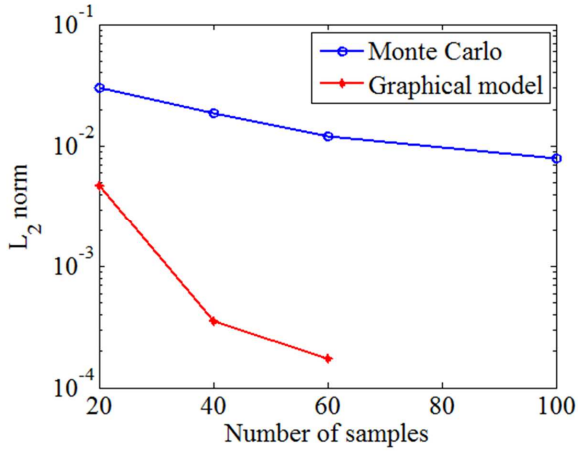

Figure 4.14: Isotropic Random Field: The L_2 norm of the error in the variance of flux as a function of the observed samples for MC simulation and graphical model prediction.

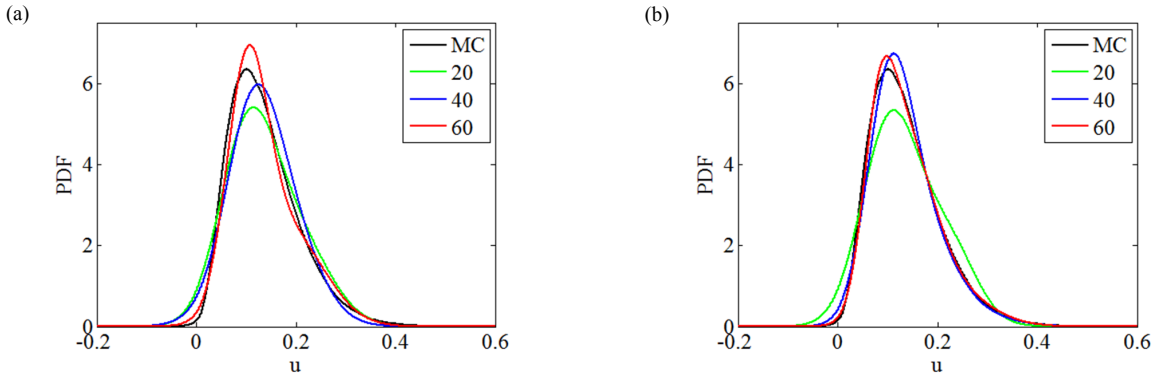

Figure 4.15: Isotropic Random Field: Predicted marginal PDF of the x-velocity at point $(0.5, 0.4375)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

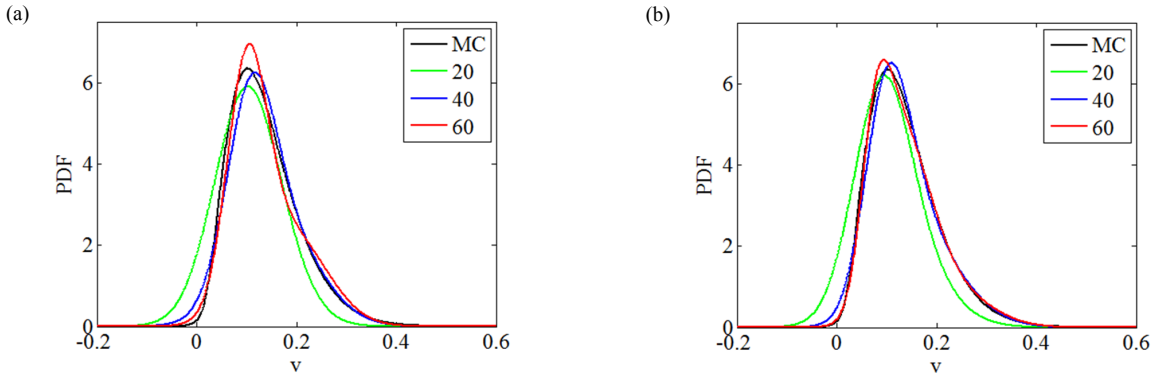

Figure 4.16: Isotropic Random Field: Predicted marginal PDF of the y-velocity at point $(0.4375, 0.5)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

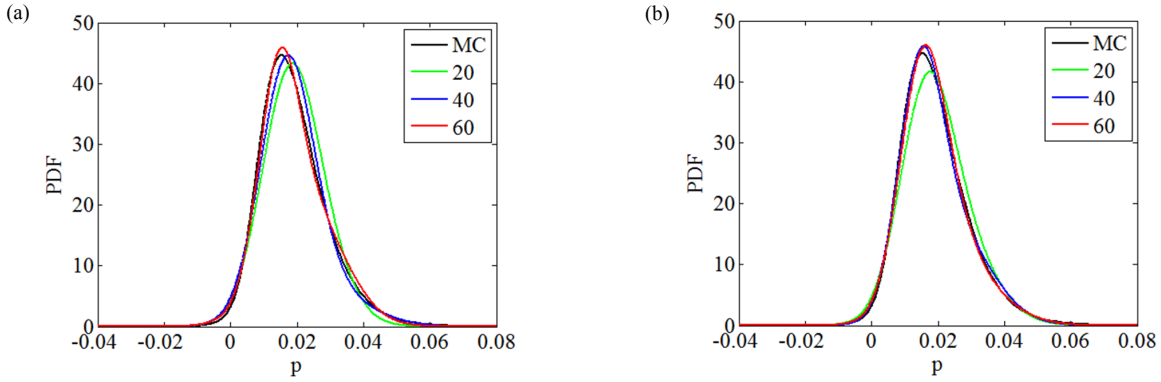

Figure 4.17: Isotropic Random Field: Predicted marginal PDF of pressure at the coarse element centered at point $(0.4375, 0.4375)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

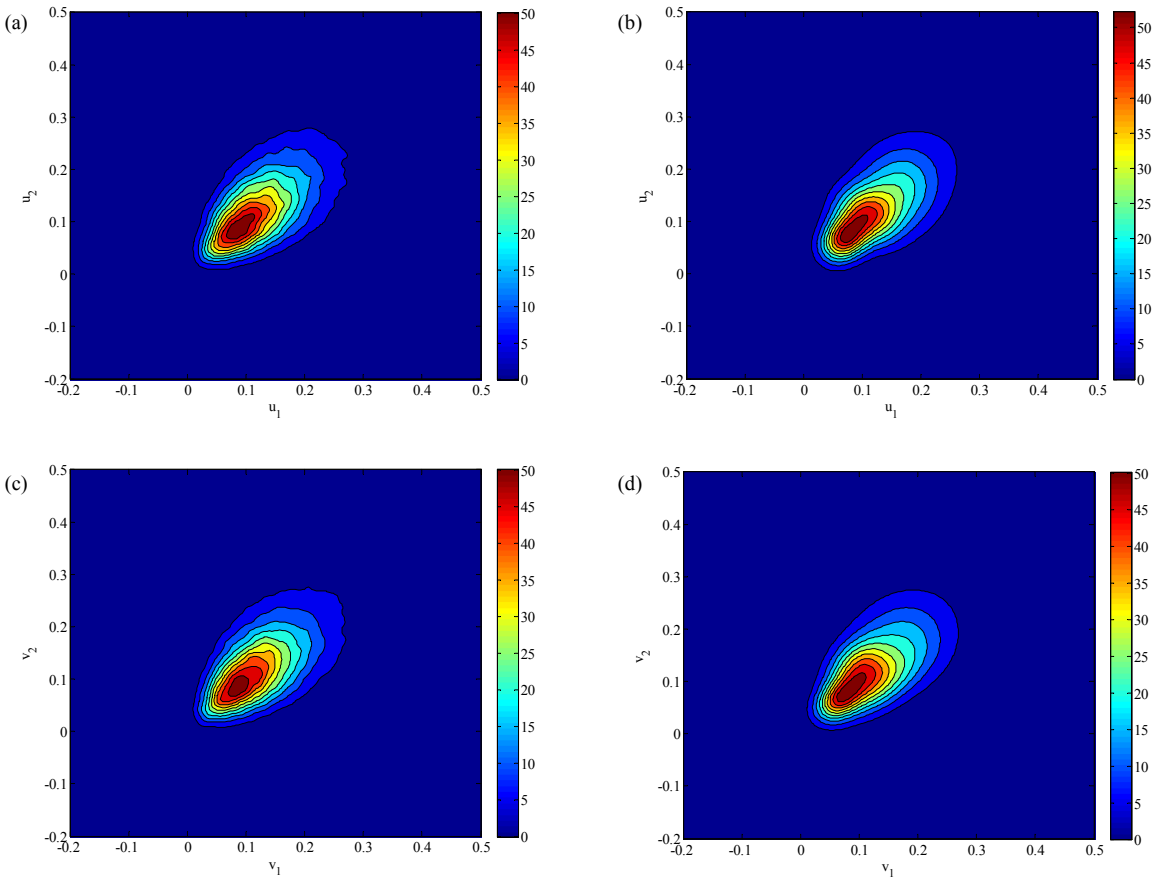

Figure 4.18: Isotropic Random Field: The joint PDF of the x-velocity u_1 at $(0.5, 0.4375)$ and u_2 at $(0.375, 0.4375)$: (a) direct simulation (b) probabilistic graphical model; the joint PDF of y-velocity v_1 at $(0.4375, 0.5)$ and v_2 at $(0.4375, 0.375)$: (c) direct simulation (d) probabilistic graphical model.

4.5.2 Anisotropic random field

In this example, an anisotropic Gaussian random field $\boldsymbol{a}$ is assumed with correlation length $L_1 = 0.1$, $L_2 = 0.2$ and standard deviation $\sigma = 1.0$. The samples of log-permeability are also generated using KL expansion with the first 100 terms. As discussed in Section 4.3, the global posterior $p(\Theta|\mathcal{D})$ can be factorized into local posterior distributions for hyperparameters related to each coarse element. Then the local hyperparameters are learned from local training data $\mathcal{D}_k$ which are local features and model responses on element E_k. This strategy relaxes the assumption of identical relationships between hidden variables and local features on different elements. However, more training data might be required to achieve sufficient accuracy in parameter learning. Here $N = 800,\ 1600,\ 2400$ training data are generated to train the probabilistic model. The source term f is set to be zero. Flow is induced from left to right side with Dirichlet boundary conditions $\bar{h} = 1$ on $x = 0$, $\bar{h} = 0$ on $x = 1$. No-flow Neumann boundary conditions are applied on the other two sides of the square domain.

As in Section 4.5.1, we first use the graphical model as a surrogate to predict model responses given randomly generated realization of the stochastic input. The probabilistic graphical model is trained with 2400 data points. The results are presented in Fig. 4.19. A 10-fold cross-validation is also performed with 1600 and 2400 samples. The cross-validation errors are shown in Fig. 4.20. Then we treat stochastic input as a random field with known probability distribution and perform belief propagation on the factor graph in Fig. 4.5(b) to predict the mean and variance of model responses. The number of kernels in nonparametric messages is set to be 4. The predictions are shown in Figs. 4.21 to 4.26. It is demonstrated in both examples that the accuracy of the probabilistic graphical

model increases with the number of training data and the predictions converge to the reference solutions. In Fig. 4.27, using MC with 10^6 samples as the reference solution, the convergence plot shows that the graphical model prediction is less accurate than MC with 200 data points. This is expected because the parameter learning process is inaccurate with too small data sets. If the estimated hyperparameters are captured on local modes of posterior distributions and significantly deviate from the true values, the probabilistic graphical model may even give incorrect predictions. However, with the increase of the number of samples, the predictions of the probabilistic graphical model quickly converge.

The marginal PDFs of the model responses are also estimated with belief propagation in Figs. 4.28, 4.29 and 4.30. Fig. 4.28 demonstrates that the accuracy of predictions can be improved by increasing the number of components in the messages in belief propagation. However, it is achieved at the expense of increased computational cost. Although too few components in Gaussian mixtures is not enough to capture the shape of non-Gaussian PDFs, using excessive Gaussian components not only increases the time for message update but may also lead to overfitting. Fig. 4.29 shows the predicted PDFs of y-velocity at point $(0.4375, 0.5)$. We can see that the empirical PDF can be efficiently captured by two Gaussian kernels in this case. Further increasing the Gaussian components does not improve significantly the accuracy of prediction. Therefore, the choice of the number of components in nonparametric belief propagation should be made by taking a balance between the computational cost and accuracy of prediction. Fig. 4.30 shows the predicted PDFs of pressure on a coarse element centered at $(0.4375, 0.4375)$. In this case, two Gaussian components are sufficient to estimate the marginal PDF accurately as it is close to Gaussian. According to these results, the proper number of Gaussian components needed in non-

parametric messages depends on the deviation of the target distribution from a Gaussian distribution. Finally, the joint distributions of x- and y-velocities at different locations are estimated with probabilistic graphical models trained by 2400 data points(Fig. 4.31).

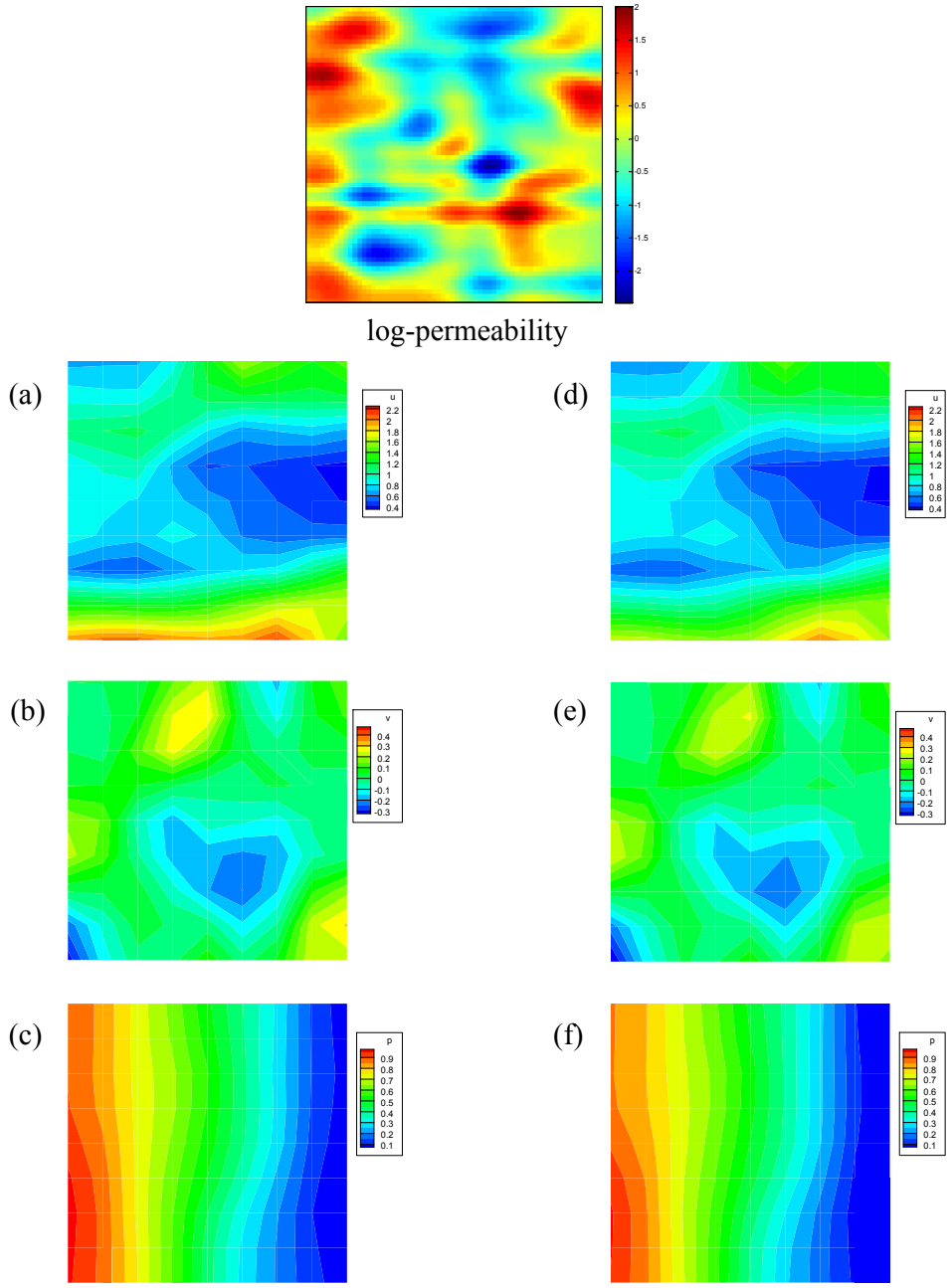

Figure 4.19: Anisotropic Random Field: Predicted values of model responses given a realization of the stochastic input (a)-(c) x-velocity, y-velocity and pressure obtained from direct simulation, and (d)-(f) x-velocity, y-velocity and pressure predicted by the probabilistic graphical model (trained with 2400 data points).

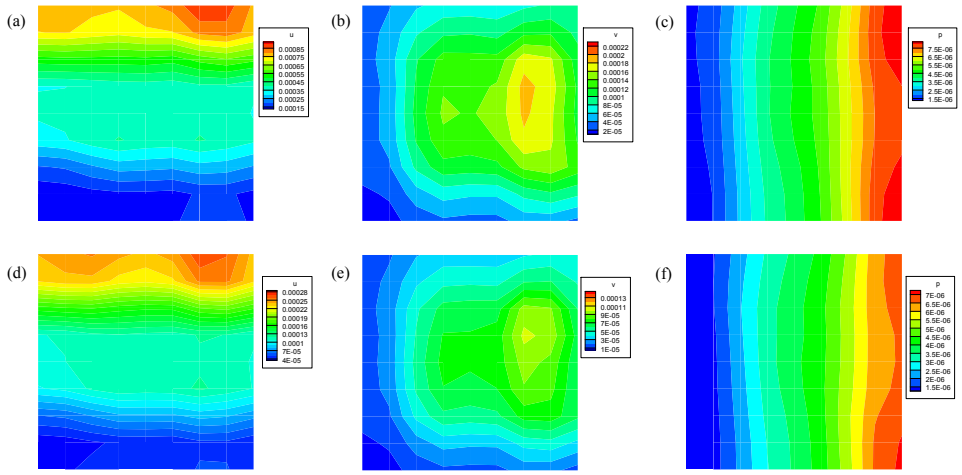

Figure 4.20: Anisotropic Random Field: k-fold cross-validation error ($k = 10$) of x-velocity, y-velocity and pressure predicted by the probabilistic graphical model with (a)-(c) 1600 samples, and (d)-(f) 2400 samples.

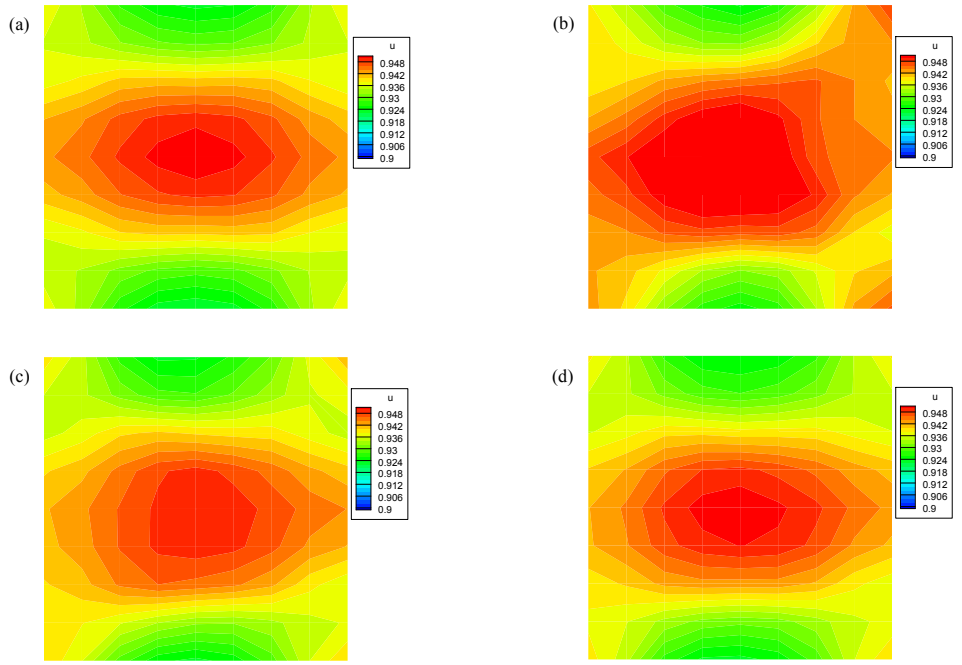

Figure 4.21: Anisotropic Random Field: Predicted mean of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

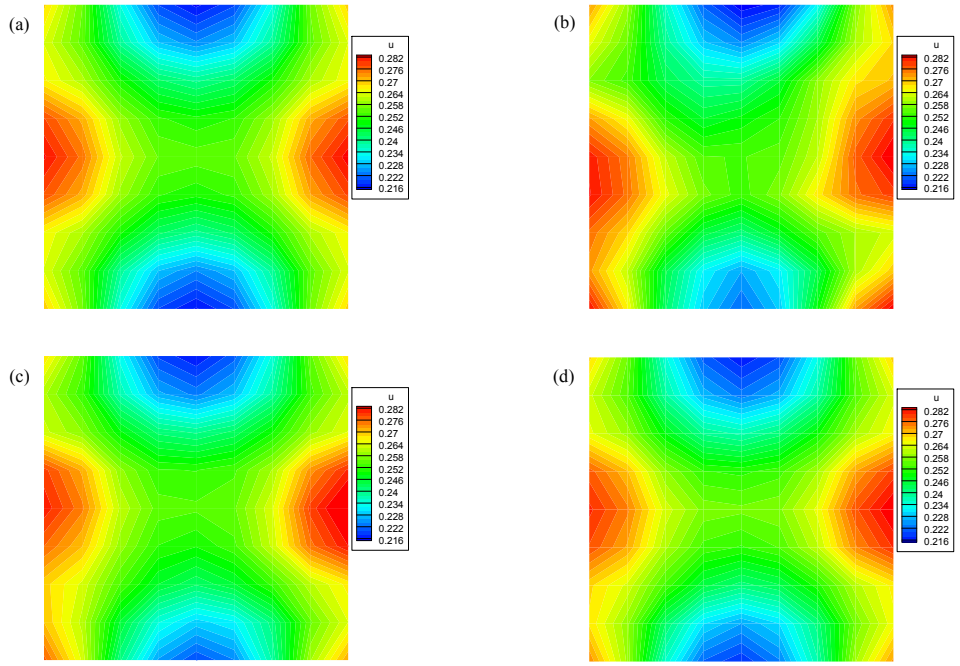

Figure 4.22: Anisotropic Random Field: Predicted variance of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

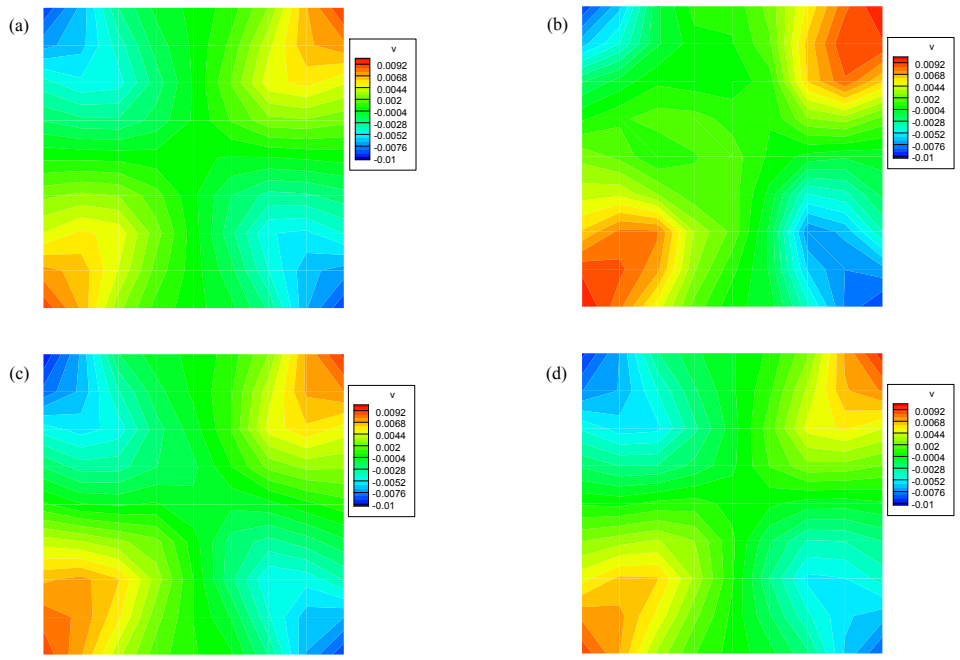

Figure 4.23: Anisotropic Random Field: Predicted mean of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

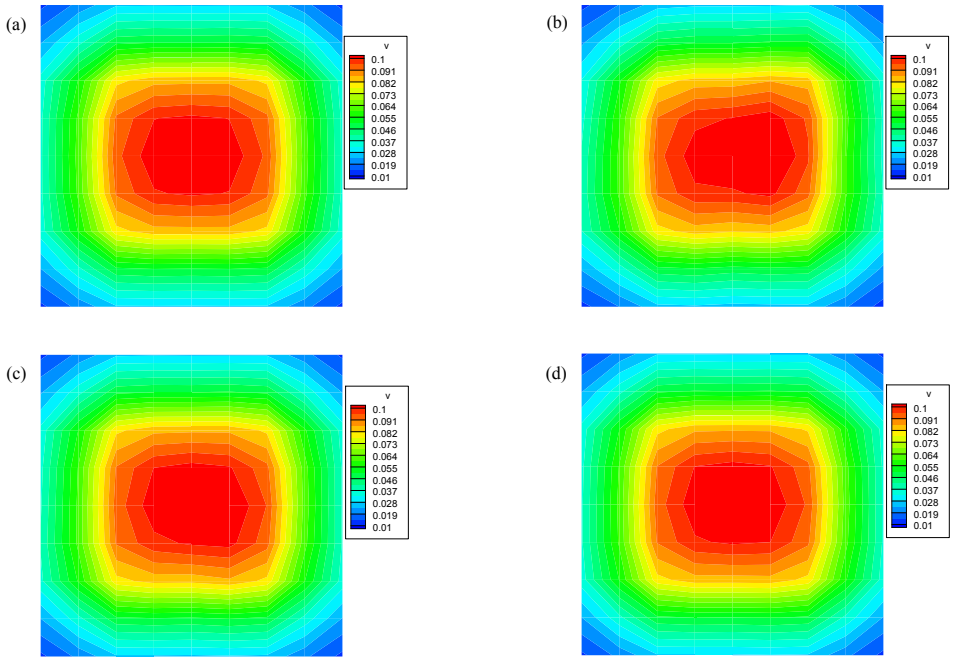

Figure 4.24: Anisotropic Random Field: Predicted variance of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

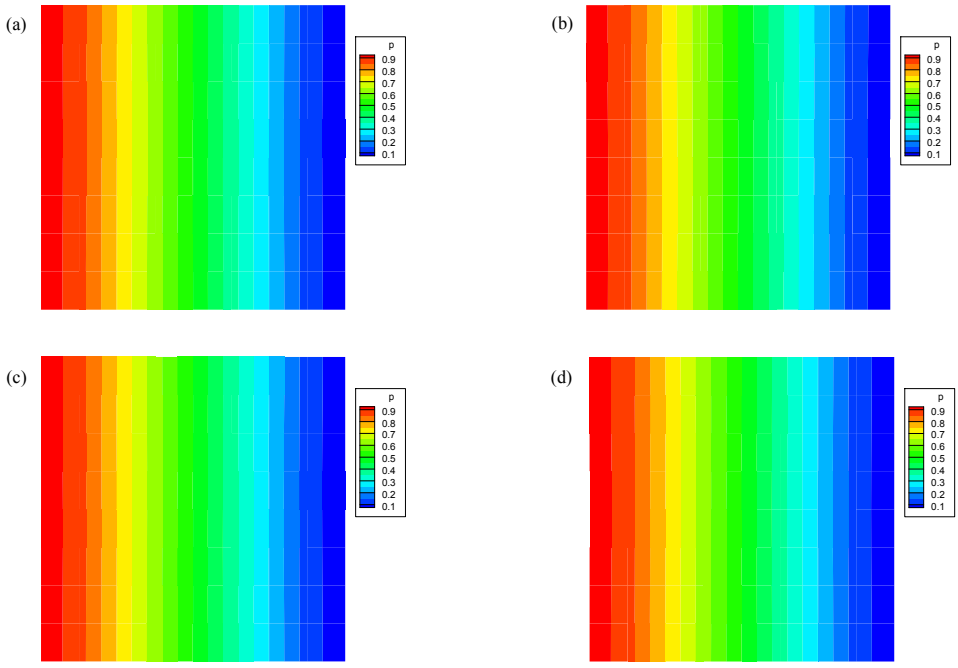

Figure 4.25: Anisotropic Random Field: Predicted mean of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

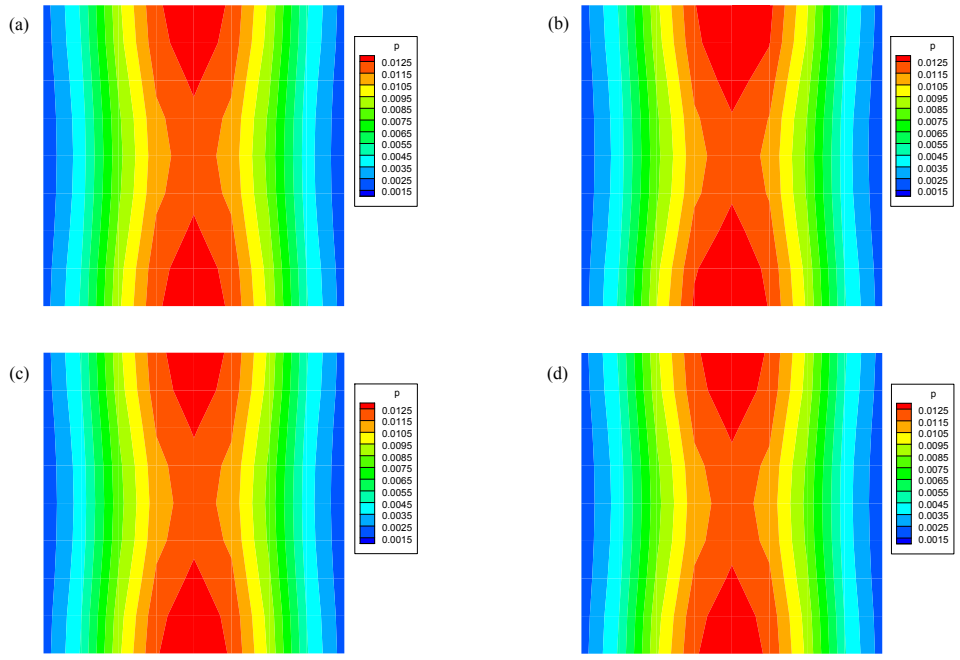

Figure 4.26: Anisotropic Random Field: Predicted variance of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

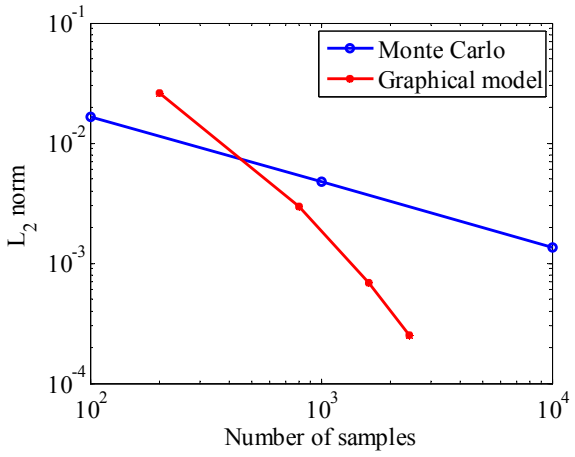

Figure 4.27: Anisotropic Random Field: The L_2 norm of the error in the variance of flux as a function of the observed samples for MC simulation and graphical model prediction.

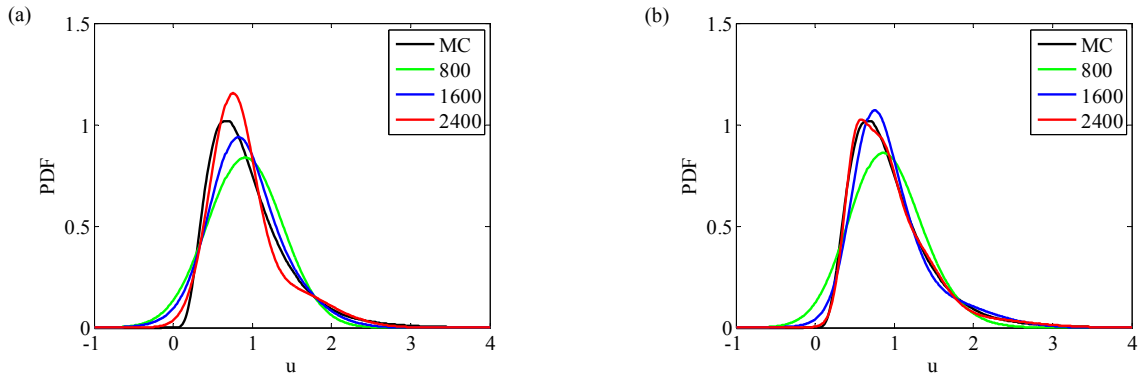

Figure 4.28: Anisotropic Random Field: Predicted marginal PDF of the x-velocity at point $(0.5, 0.4375)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

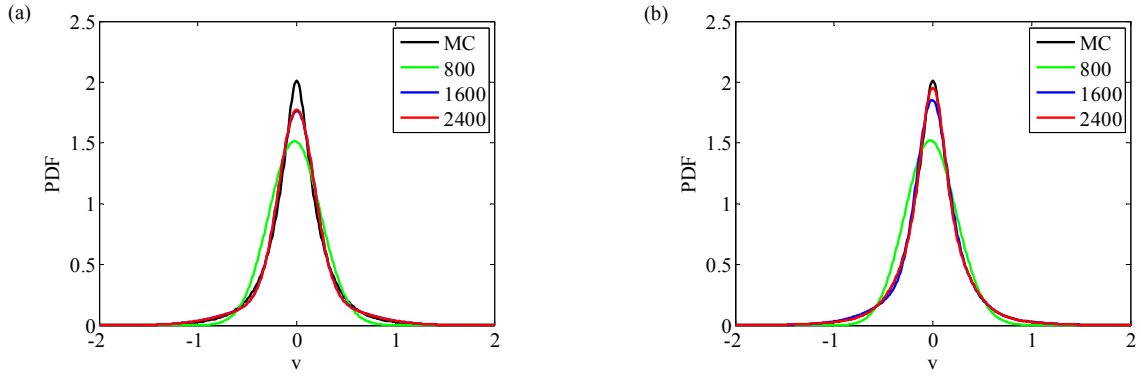

Figure 4.29: Anisotropic Random Field: Predicted marginal PDF of the y-velocity at point $(0.4375, 0.5)$: (a) 2 and (b) 4 Gaussian components in nonparametric messages.

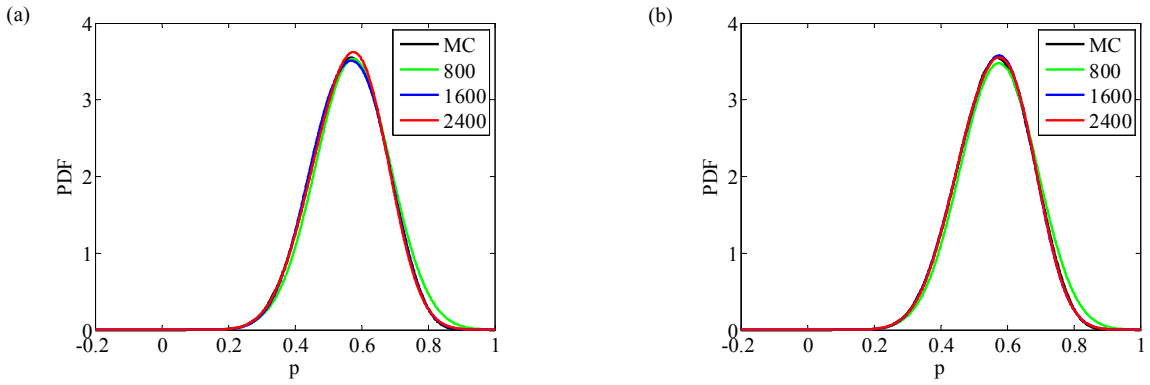

Figure 4.30: Anisotropic Random Field: Predicted marginal PDF of pressure at the coarse element centered at point $(0.4375, 0.4375)$: (a) 2 and (b) 4 Gaussian components in nonparametric messages.

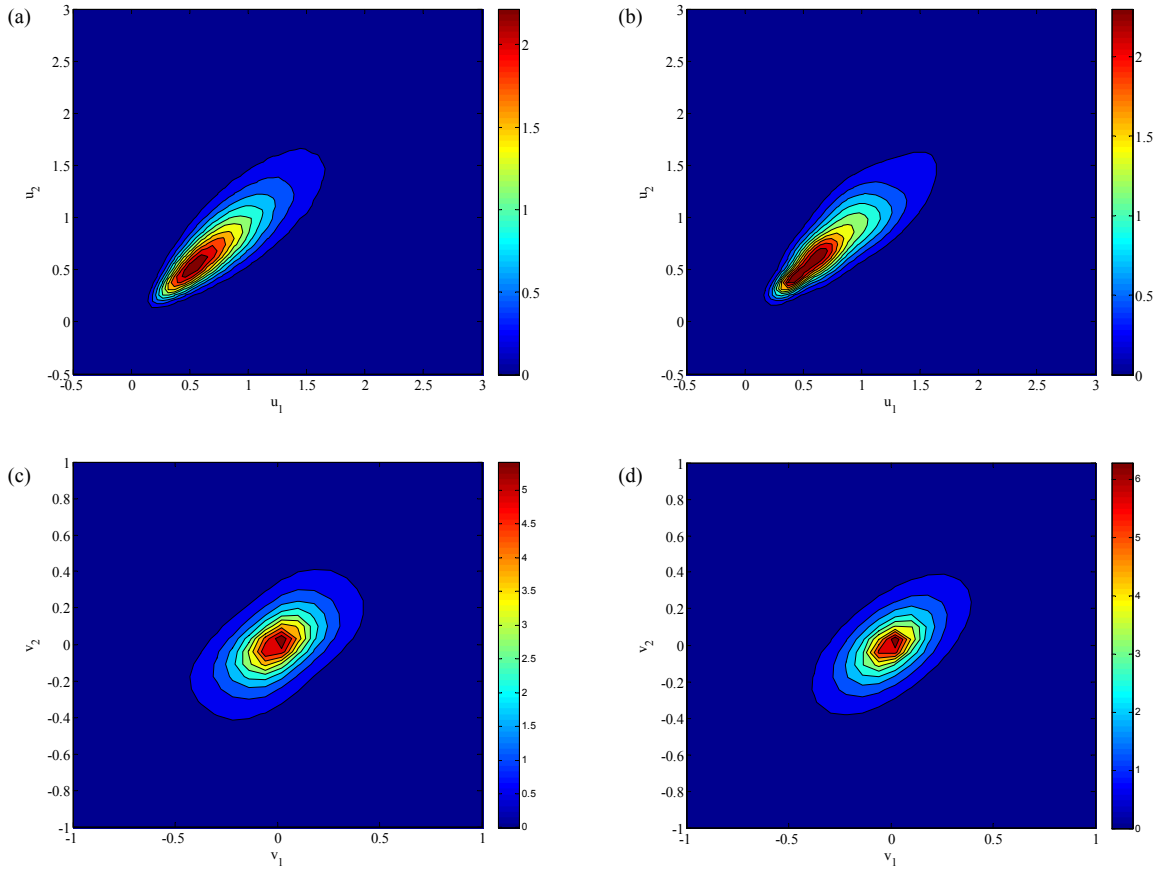

Figure 4.31: Anisotropic Random Field: The joint PDF of the x-velocity u_1 at $(0.5, 0.4375)$ and u_2 at $(0.375, 0.4375)$: (a) direct simulation (b) probabilistic graphical model; the joint PDF of y-velocity v_1 at $(0.4375, 0.5)$ and v_2 at $(0.4375, 0.375)$: (c) direct simulation (d) probabilistic graphical model.

4.5.3 Nonstationary random field

In the previous examples, it was assumed that the porous media are stationary such that the covariance between any two points in the domain depends on their distance rather than their actual locations. However, hydraulic properties may exhibit spatial variations at various scales. Therefore, it is important to extend the probabilistic graphical model to nonstationary random fields. In this example, we use a nonstationary random field as stochastic input. The log-permeability on the k-th coarse element is a Gaussian random field with mean zero and an exponential covariance function

$$\mathrm{cov}(\boldsymbol{x}, \boldsymbol{x}^\star) = \sigma^2 \exp(-\frac{|x_1 - x_1^\star|}{L_{k,1}} - \frac{|x_2 - x_2^\star|}{L_{k,2}}). \tag{4.39}$$

The correlation lengths $L_{k,1}$ and $L_{k,2}$ vary on the coarse scale. Since the coarse grid has $N_x = 8$ rows and $N_y = 8$ columns of elements, we define the coordinate of the k-th element as (i_k, j_k) where i_k is the index in row and j_k is the index in column. Then the correlation length is set to be $L_{k,1} = 0.1 + \frac{0.4}{N_y - 1} j_k$ and $L_{k,2} = 0.1 + \frac{0.4}{N_x - 1} i_k$. The source term and boundary conditions are the same as those in Section 4.5.2.

The challenge of a nonstationary random field is that the influence of local properties on local responses could vary on coarse elements as the correlation between hydraulic properties depends on the location. In this case, it is difficult to estimate the probabilistic model globally due to the large number of hyperparameters. The probabilistic graphical model proposed in Section 4.2.2 efficiently decomposes the global problem into local lower dimensional problems. In this way, we can estimate the hyperparameters locally with local posterior distributions defined in Section 4.3.

In this example, $N = 800,\ 1600,\ 2400$ training data are generated to train the probabilistic model. The belief propagation runs in the same way as in previous examples. Fig. 4.32 and 4.33 verifies the correctness of the probabilistic model in its ability as a surrogate model. The estimated mean and variance of model responses are shown in Figs. 4.34 to 4.39. The convergence plot is presented in Fig. 4.40. Compared with the convergence plot in the example of Section 4.5.2, it is seen that the convergence rate is not significantly affected by the stationarity/nonstationarity. This is because the hyperparameters are learned locally, which implies that the convergence depends on the number of data sets and the number of hyperparameters in each coarse element. Figs. 4.41 to 4.43 show the predicted marginal PDF of model responses. Fig. 4.44 shows the joint distributions of model responses at different locations estimated with a probabilistic graphical model trained by 2400 data points. With the local hyperparameter learning strategy, this example requires no more training data than the last example in which the stochastic input is a stationary random field. On the other hand, it is suggested to use global parameter learning for stationary random field as hyperparameters on different coarse elements are identical. According to the first example in Section 4.5.1, a relatively small number of training data is sufficient to give accurate probabilistic models.

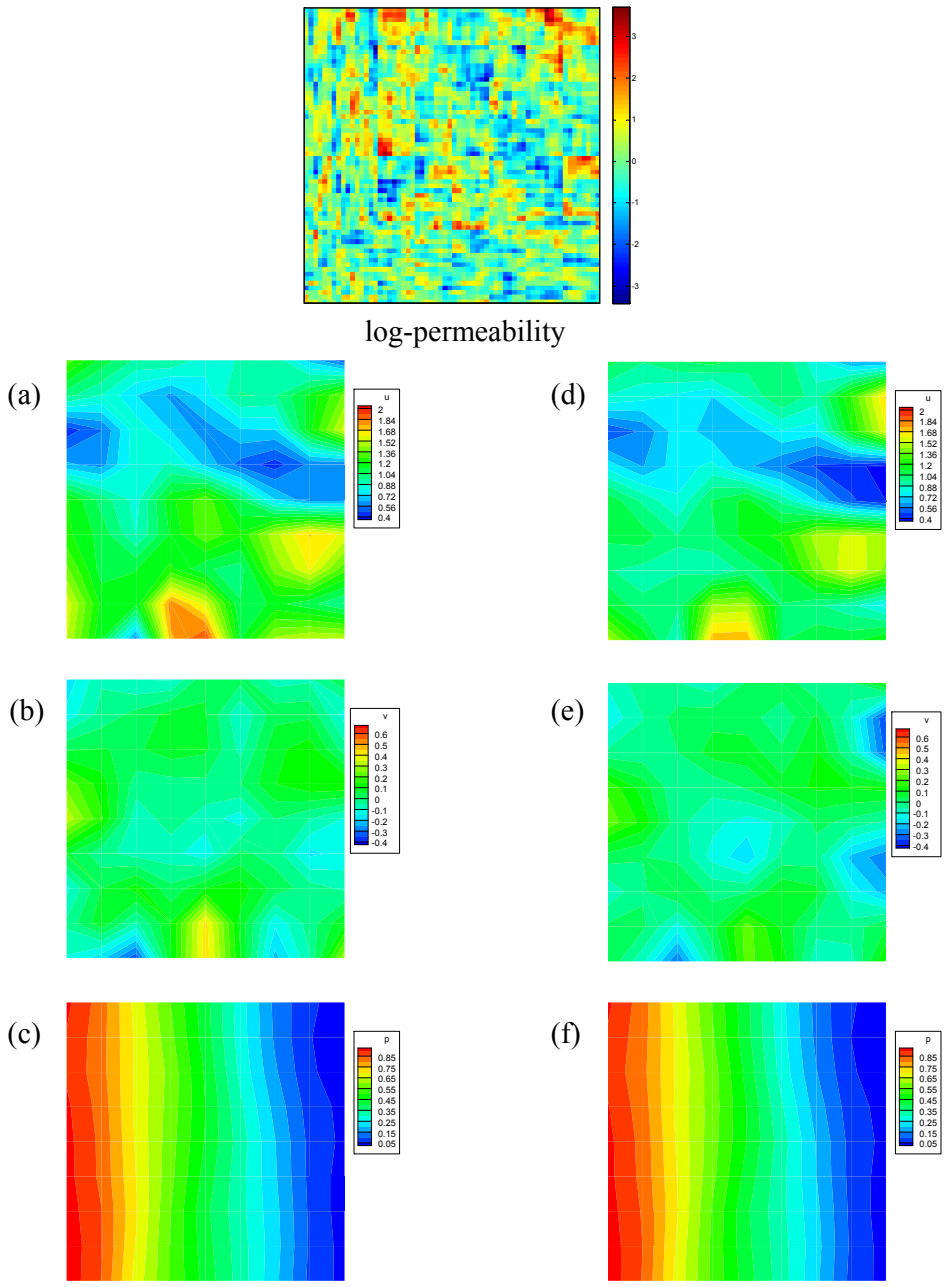

Figure 4.32: Nonstationary Random Field: Predicted values of model responses given a realization of stochastic input (a)-(c) x-velocity, y-velocity and pressure obtained from direct simulation, and (d)-(f) x-velocity, y-velocity and pressure predicted by the probabilistic graphical model (trained with 2400 data points).

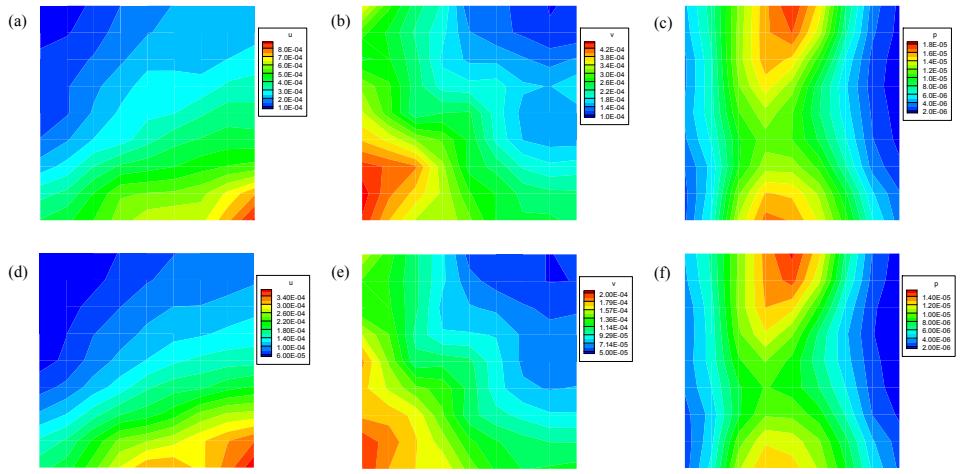

Figure 4.33: Nonstationary Random Field: k-fold cross-validation error $(k = 10)$ of x-velocity, y-velocity and pressure predicted by the probabilistic graphical model with (a)-(c) 1600 samples, and (d)-(f) 2400 samples.

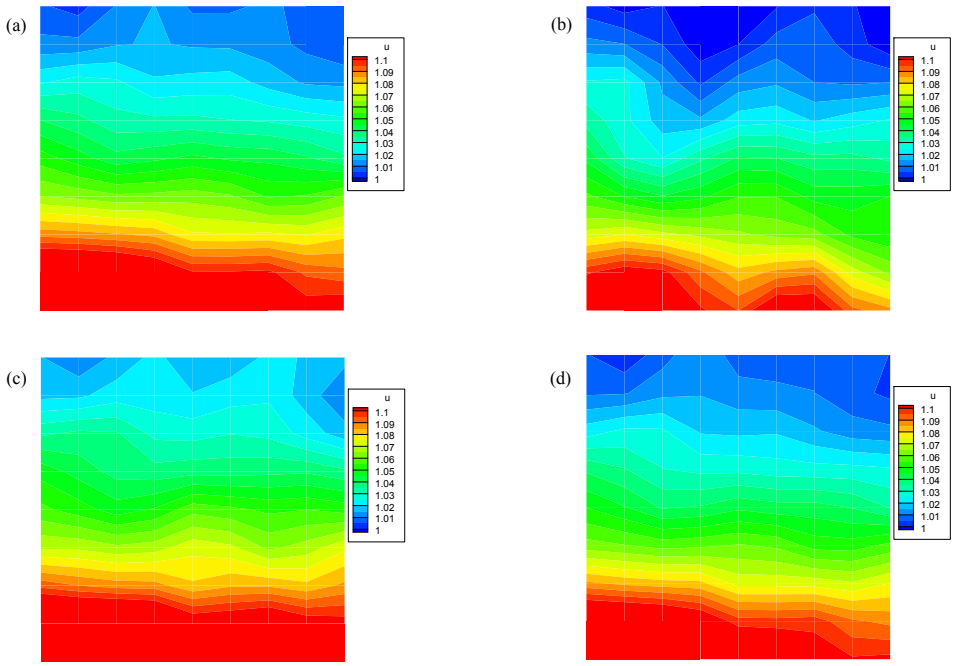

Figure 4.34: Nonstationary Random Field: Predicted mean of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

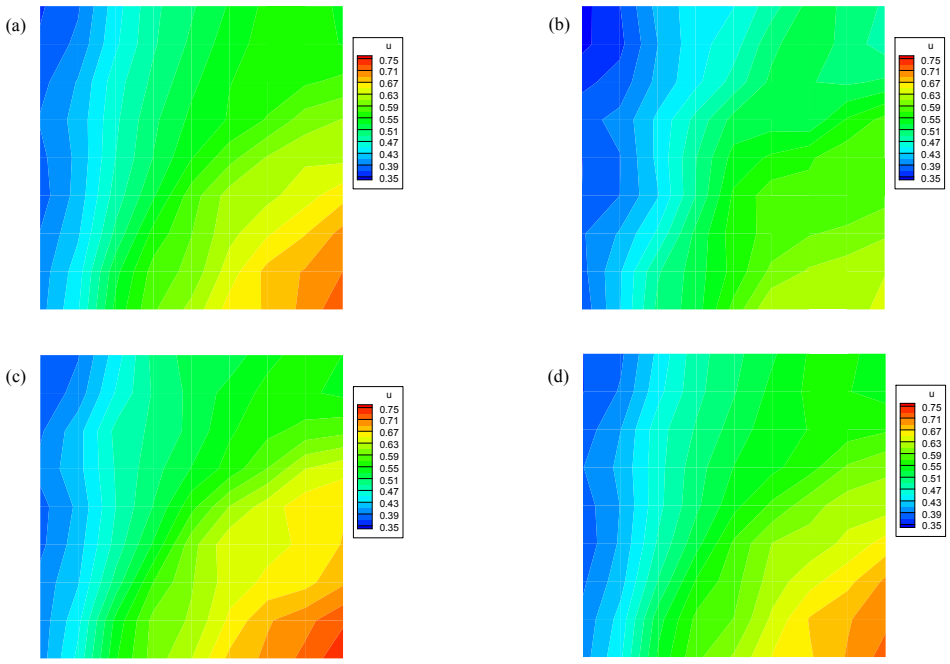

Figure 4.35: Nonstationary Random Field: Predicted variance of the x-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

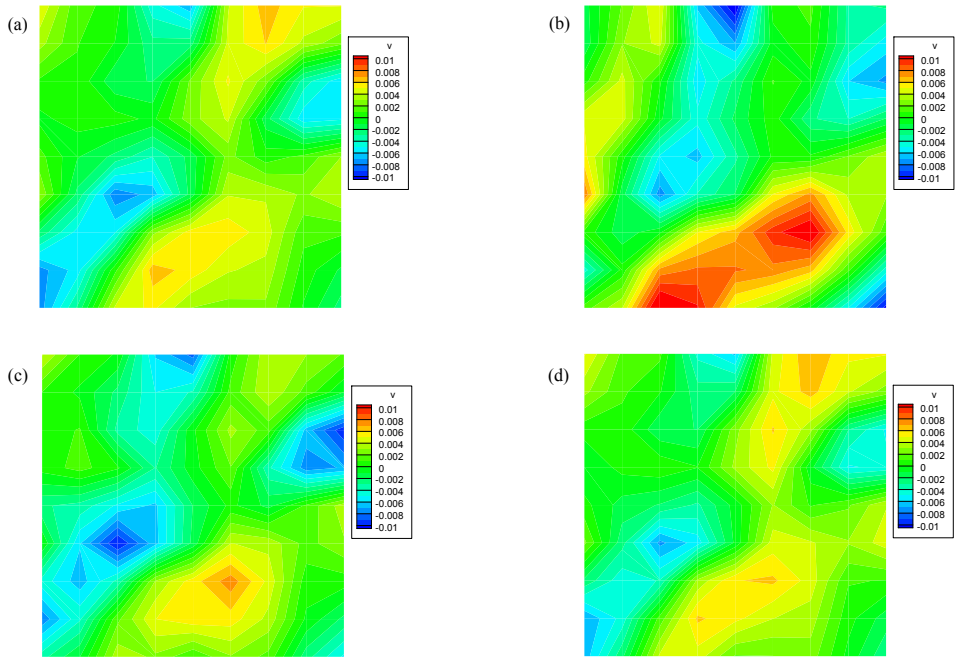

Figure 4.36: Nonstationary Random Field: Predicted mean of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

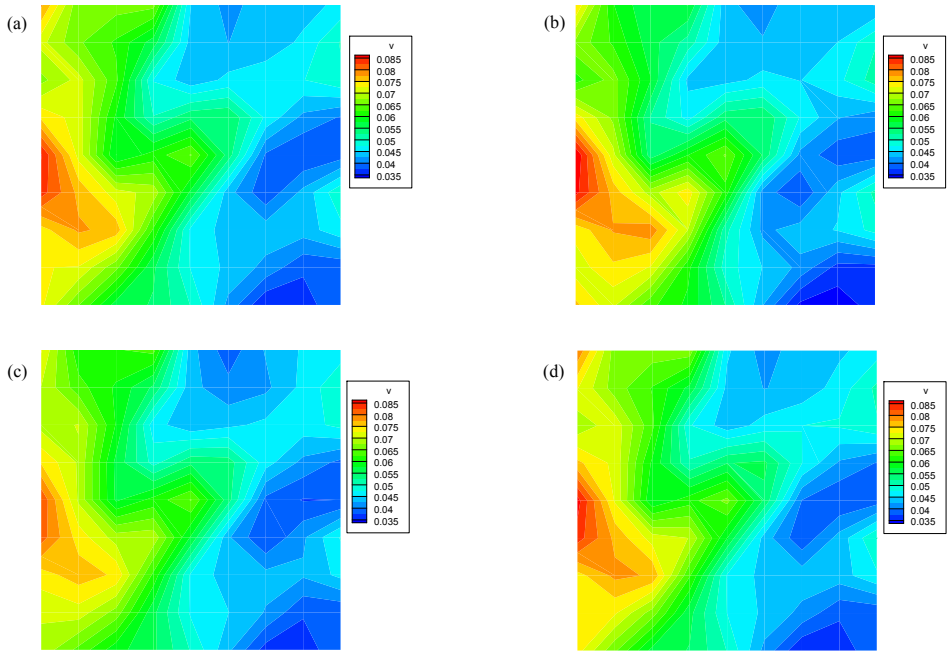

Figure 4.37: Nonstationary Random Field: Predicted variance of the y-velocity from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

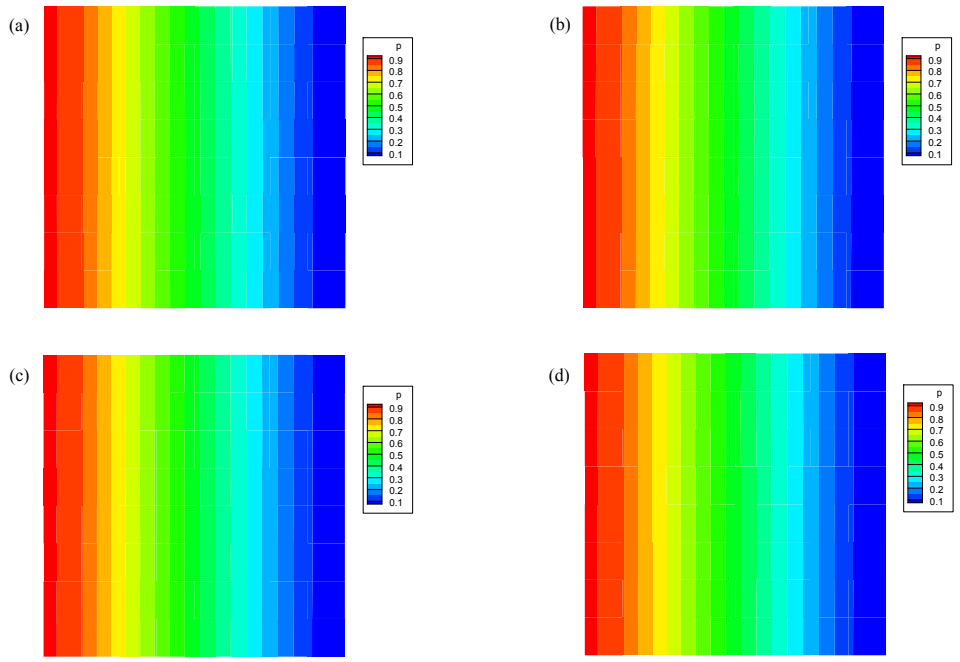

Figure 4.38: Nonstationary Random Field: Predicted mean of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

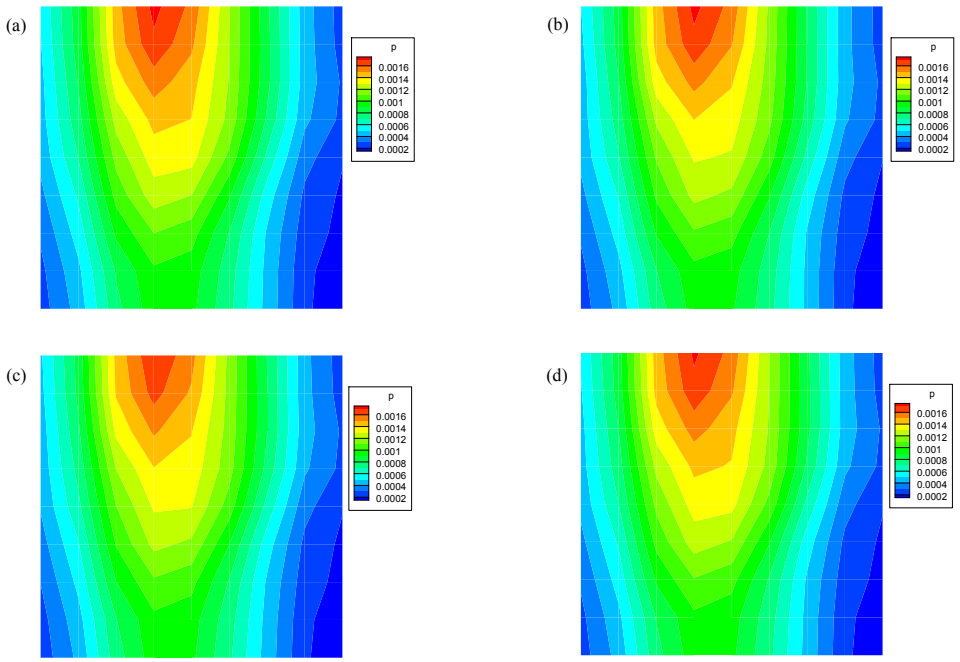

Figure 4.39: Nonstationary Random Field: Predicted variance of pressure from (a) MC simulation, and from probabilistic graphical models trained by (b) 800, (c) 1600 and (d) 2400 data.

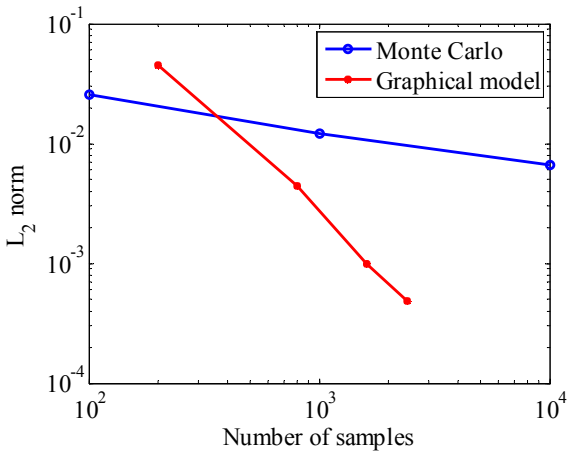

Figure 4.40: Nonstationary Random Field: The L_2 norm of the error in the variance of flux as a function of the observed samples for MC simulation and graphical model prediction.

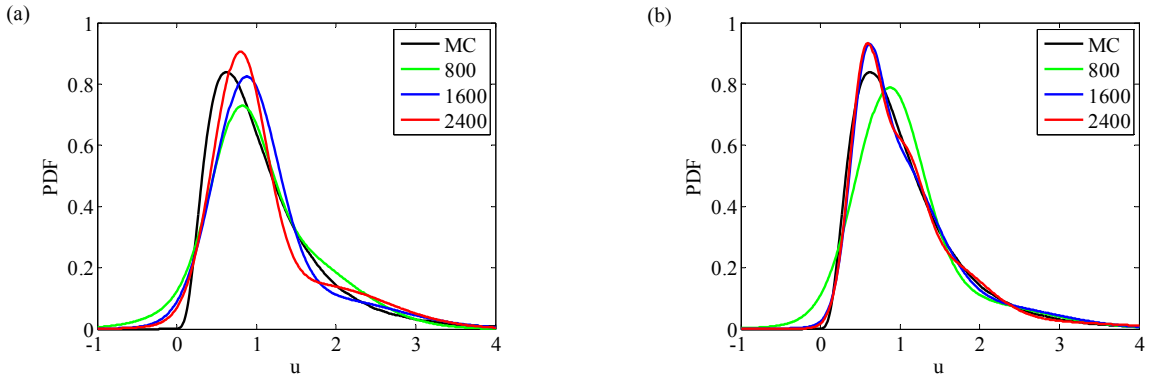

Figure 4.41: Nonstationary Random Field: Predicted marginal PDF of the x-velocity at point $(0.5, 0.4375)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

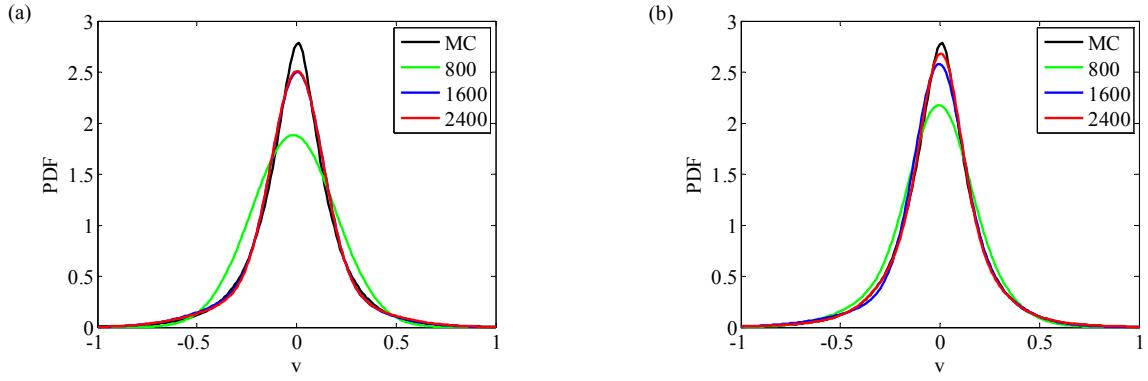

Figure 4.42: Nonstationary Random Field: Predicted marginal PDF of the y-velocity at point $(0.4375, 0.5)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

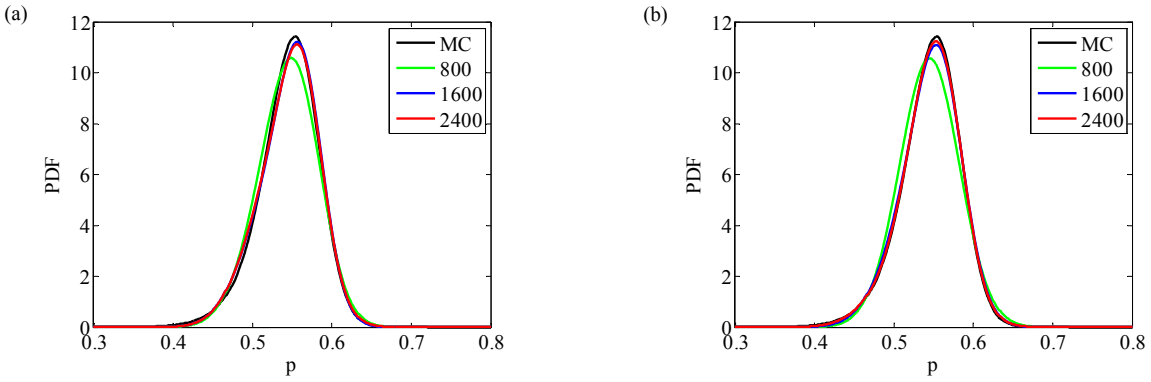

Figure 4.43: Nonstationary Random Field: Predicted marginal PDF of pressure at the coarse element centered at point $(0.4375, 0.4375)$: Using (a) 2 and (b) 4 Gaussian components in nonparametric messages.

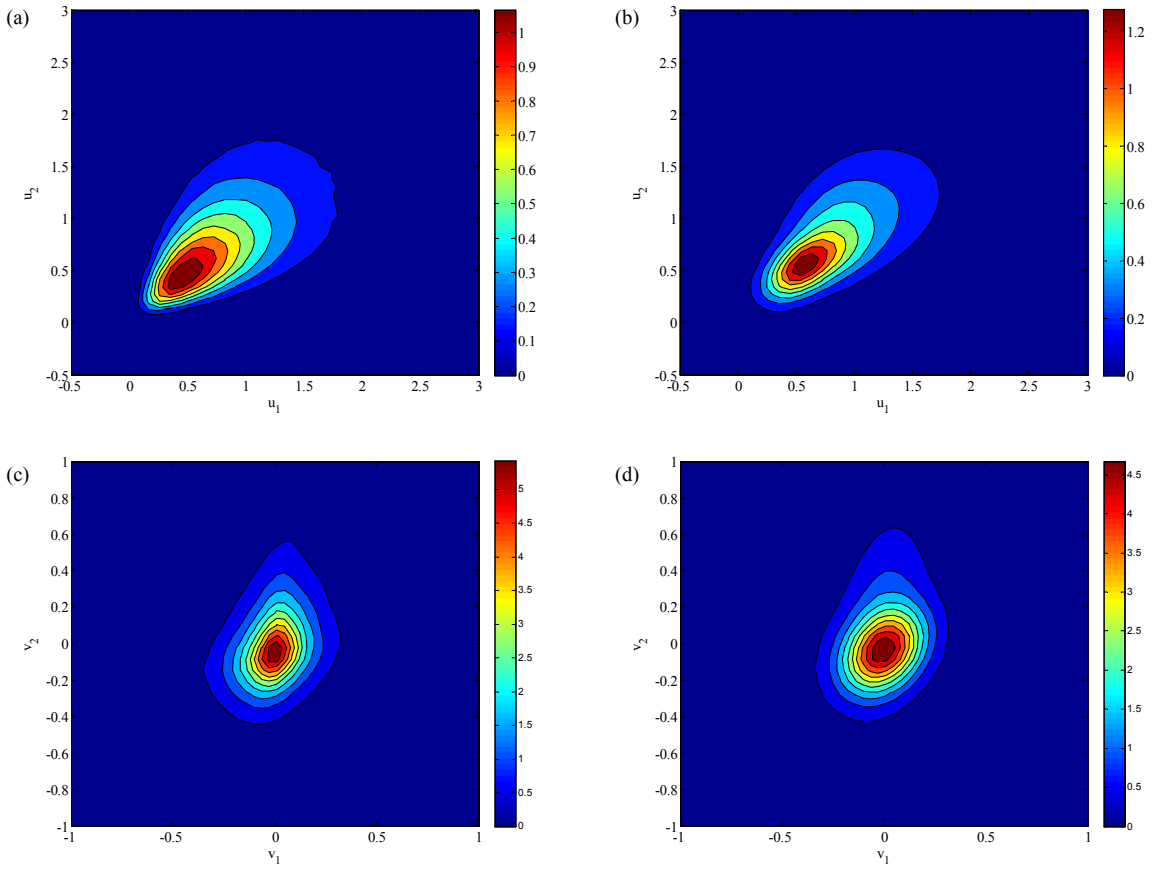

Figure 4.44: Nonstationary Random Field: The joint PDF of the x-velocity u_1 at $(0.5, 0.4375)$ and u_2 at $(0.375, 0.4375)$: (a) direct simulation (b) probabilistic graphical model; the joint PDF of the y-velocity v_1 at $(0.4375, 0.5)$ and v_2 at $(0.4375, 0.375)$: (c) direct simulation (d) probabilistic graphical model.

4.6 Conclusions

In this chapter, a probabilistic graphical model is constructed to approximate the solutions to multiscale SPDEs. The basic idea is to treat the stochastic input as well as model responses as random variables and to represent their relationships explicitly using a factor graph model. Thus the high-dimensional joint distribution can be factorized by the potential functions that describe the interactions among neighboring variables in the graph. In order to relieve the curse of dimensionality, a set of hidden variables defined on a coarse-scale are employed to bridge fine-scale features and coarse-scale responses. The graphical model not only facilitates probabilistic modeling of model responses but also enables us to solve the inference problem efficiently with the help of belief propagation algorithms. These algorithms marginalize unobserved random variables in a graphical model by propagating special functions, messages, between variables (including variable nodes and factor nodes in a factor graph).

One of the most important issues in the belief propagation algorithm with nonparametric messages is the specification of number of Gaussian components in the nonparametric messages. Numerical results show that insufficient number of components will lead to inaccurate predictions, while excessive components will increase computational cost or even lead to overfitting. The proper number of components mainly depends on the deviation of target marginal distributions from Gaussian distributions.

It is also straightforward to extend the current graphical model to incorporate higher-order interactions between random variables. Note that the interactions between variables are explicitly denoted by the factor nodes in the factor

graph in Fig. 4.3(b). Each factor node is a potential function and connects all its member variable nodes. When a high-order potential function is considered, one just adds a corresponding factor node in the factor graph and connects it with all its member variables. The belief propagation algorithm introduced in Section 4.4 strictly follows the update rules in Eq. (4.28) and (4.29) which in theory can deal with any-order potential functions. Thus we can use the same algorithm to make inference on a factor graph with high-order potentials. However, high-order interactions can lead to more complicated graph structure and high-dimensional message propagation. As a result, the computational cost would be significantly increased.

CHAPTER 5

CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH

In this thesis, we developed efficient computational techniques for three important problems related to high-dimensional stochastic modeling: (1) inverse problems with high-dimensional input, (2) stochastic input model construction and (3) uncertainty quantification in multiscale systems with high-dimensional input. A common challenge in all three problems is the *curse of dimensionality* that is addressed in each problem using different strategies. The basic idea of dealing with high-dimensional spatially distributed input in inverse problems is to represent it hierarchically. Then an adaptive hierarchical sampling strategy can be applied. This framework is very efficient especially when little prior information on the input is available and there exists discontinuity in the spatially distributed input. Another novelty of our work is the application of the probabilistic graphical model approach to uncertainty quantification. In the construction of the stochastic input model from observation data, a Bayesian network is utilized to factorize a high-dimension joint distribution into product of lower-dimensional conditional distributions. As a result, conventional density estimators, which are limited to low-dimensional problems, can be employed. This idea is further extended to give a non-sampling approach for uncertainty quantification of multiscale systems. When stochastic input and model responses are treated as random variables, a probabilistic graphical model for a multiscale system can be constructed based on dependence relationships between these random variables. Then efficient inference algorithms are applied to make inference of unobserved variables, i.e. model responses, directly without involving sampling and expensive deterministic solvers.

The efficiency of computational techniques developed in this thesis has been demonstrated with numerical examples. However, there are still several areas where further developments are required. Suggestions for the continuation of this study are provided next.

5.1 Constructing probabilistic graphical models for multiscale systems

The construction of a graphical model is based on knowledge of dependence relationships between random variables including stochastic input $\boldsymbol{a}$ and model responses $\boldsymbol{Y}$. However, such relationships are often implicit and are not easy to identify. There are two promising ways of addressing this issue:

(1) Obtain dependence relationships from homogenization theories. Generally, fine-scale features are homogenized to coarse-scale variables and the system is solved on a coarse-scale to predict model responses. Consequently, the relationships between input and output variables are embedded in the deterministic solver. A homogenization method defines the way input, output as well as coarse-grained variables are correlated. Thus it is possible to translate the underlying dependence relationships into a graph configuration using graph theories. In this thesis, we have successfully constructed a graphical model from mixed multiscale finite element method in this way. Actually, various more generalized homogenization methods, e.g. heterogeneous multiscale method, can be utilized to construct different probabilistic graphical models for particular systems.

(2) The other choice is to learn the graph structure directly from observation data. The main challenge of this task is that the number of candidate graphs increases exponentially with the number of random variables. Two categories of graphical structure learning algorithms have been developed in past decades: search-and-score and constraint-based methods. The former assigns a score to each possible graphical model and find one that maximizes the score given observation data. The latter learns graph structures by running local conditional independence tests to identify a model containing independence relationships among random variables. Systematical studies on both categories of methods are required to find appropriate learning algorithms which take a balance between complexity and accuracy in approximating $p(\boldsymbol{Y}|\boldsymbol{a})$ in a medium or high-dimensional space.

5.2 Application of probabilistic graphical model to inverse problems in the multiscale context

According to Bayes rule, $p(\boldsymbol{a}|\boldsymbol{Y}) = p(\boldsymbol{Y}|\boldsymbol{a})p(\boldsymbol{a})$. Given prior distribution of stochastic input $p(\boldsymbol{a})$, the probabilistic graphical model of $p(\boldsymbol{Y}|\boldsymbol{a})$ can also be used to infer the fine-scale input given an observation of coarse-scale model responses. The major issues to be solved include: (1) the design of a proper graph structure. While stochastic input and response variables are fixed, the choice of hidden variables (i.e. coarse-grained variables) is flexible. As inverse problems are often ill-posed, a well-designed graph structure may improve the regularization. We need to conduct rigorous investigations on the influence of graphical model representation on the regularization of multiscale inverse prob-

lems. (2) an efficient inference algorithm for joint distributions of unobserved random variables. In theory, current inference algorithms can directly predict the posterior distribution from the graph. However, many existing algorithms, e.g. loopy belief propagation, are only efficient in obtaining the marginal probability density functions of stochastic input (with polynomial complexity). It is still challenging to accurately approximate the multivariate distributions of arbitrary random variables from the graph due to the curse of dimensionality.

BIBLIOGRAPHY

[1] J.E. Aarnes, V. Kippe, K-A Lie, and A.B. Rustad. Modelling of Multiscale Structures in Flow Simulations for Petroleum Reservoirs. In G. Hasle, K-A Lie, and E. Quak, editors, *Geometric Modelling, Numerical Simulation, and Optimization*, chapter 10, pages 307–360. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[2] J.E. Aarnes, V. Kippe, K-A Lie, and A.B. Rustad. Modelling of Multiscale Structures in Flow Simulations for Petroleum Reservoirs. In G. Hasle, K-A Lie, and E. Quak, editors, *Geometric Modelling, Numerical Simulation, and Optimization*, chapter 10, pages 307–360. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[3] D.N. Arnold. Mixed finite element methods for elliptic problems. *Comput. Methods Appl. Mech. Engrg.*, 82(1-3):281–300, 1990.

[4] J. Besag. Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236, 1974.

[5] J. Besag, P. Green, D. Higdon, and K. Mengersen. Bayesian computation and stochastic systems. *Statistical Science*, 10(1):3–41, 1995.

[6] A. Billy, F.Y. Bois, E. Parent, and C.P. Robert. Bayesian-optimal design via interacting particle systems. *Journal of the American Statistical Association*, 101(474):773–785, 2006.

[7] J. Bilmes. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical report, 1998.

[8] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[9] C.M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st ed. 2006. corr. 2nd printing edition, October 2007.

[10] T. Bouezmarni, J. V. K. Rombouts, and A. Taamouti. A nonparametric copula based test for conditional independence with applications to granger causality. Economics Working Papers we093419, 2009.

[11] H-J Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:147 – 269, 2004.

[12] D. Calvetti and E. Somersalo. *Introduction to Bayesian Scientific Computing : Ten Lectures on Subjective Computing*. Springer, 2007.

[13] S. Chakraborty. Some applications of Dirac's delta function in statistics for more than one random variable. *Appl. Appl. Math.*, 3(1):42–54, 2008.

[14] J. Cheng, R. Greiner, J. Kelly, D. Bell, and W. Liu. Learning bayesian networks from data: an information-theory based approach. *Artif. Intell.*, 137:43–90, May 2002.

[15] C.P. de Campos, Z. Zeng, and Q. Ji. Structure learning of Bayesian networks using constraints. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 113–120, New York, NY, USA, 2009. ACM.

[16] A. Doostan and G. Iaccarino. A least-squares approximation of partial differential equations with high-dimensional random inputs. *J. Comput. Phys.*, 228(12):4332–4345, July 2009.

[17] P. Dostert, Y. Efendiev, and T.Y. Hou. Multiscale finite element methods for stochastic porous media flow equations and application to uncertainty quantification. *Computer Methods in Applied Mechanics and Engineering*, 197:3445–3455, 2008.

[18] P. Dostert, Y. Efendiev, and B. Mohanty. Efficient uncertainty quantification techniques in inverse problems for richards equation using coarse-scale simulation models. *Advances in Water Resources*, 32(3):329–339, 2009.

[19] A. Doucet, N. De Freitas, and N. Gordon, editors. *Sequential Monte Carlo methods in practice*. Springer-Verlag, 2001.

[20] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[21] W. E and B. Engquist. The heterogeneous multi-scale methods. *Comm. Math. Sci*, 1:87–132, 2002.

[22] W. E, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden. Heterogeneous

multiscale methods: A review. *Communications in Computational Physics*, 2:367–450, 2007.

[23] D. I. Edwards. *Introduction to Graphical Modelling*. Springer, 2nd edition, 2000.

[24] Y. Efendiev, J. Galvis, and P. Vassilevski. Spectral element agglomerate algebraic multigrid methods for elliptic problems with high-contrast coefficients. In Yunqing Huang, Ralf Kornhuber, Olof Widlund, and Jinchao Xu, editors, *Domain Decomposition Methods in Science and Engineering XIX*, volume 78 of *Lecture Notes in Computational Science and Engineering*, pages 407–414. Springer Berlin Heidelberg, 2011.

[25] Y. Efendiev and T.Y. Hou. *Multiscale Finite Element Methods: Theory and Applications*. Surveys and Tutorials in the Applied Mathematical Sciences. Springer, Dordrecht, 2009.

[26] M.A.R. Ferreira and H.K.H. Lee. *Multiscale Modeling: A Bayesian Perspective*. Springer, 2007.

[27] R. Ghanem and P. D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer - Verlag, New York, 1991.

[28] P.J. Green. Reversible jump markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[29] M. Haran. *Gaussian random field models for spatial data. In Markov chain Monte Carlo handbook Eds. Brooks,S.R., Gelman, Andrew, Jones, G.L. and Meng, X.L. (to appear)*.

[30] D. Heckerman. A tutorial on learning with bayesian networks. In Dawn Holmes and Lakhmi Jain, editors, *Innovations in Bayesian Networks*, volume 156 of *Studies in Computational Intelligence*, pages 33–82. Springer Berlin / Heidelberg, 2008.

[31] D. Heckerman, D. Geiger, and D.M. Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

[32] J.S. Hesthaven and E.M. Rønquist. *Spectral and High Order Methods for Partial Differential Equations: Selected Papers from the ICOSAHOM '09 Con-*

ference, June 22-26, Trondheim, Norway. Lecture Notes in Computational Science and Engineering. Springer, 2010.

[33] D. Higdon, C. Nakhleh, J. Gattiker, and B. Williams. A Bayesian calibration approach to the thermal problem. *Computer Methods in Applied Mechanics and Engineering*, 197(29-32):2431–2441, 2008.

[34] T. Y. Hou and X. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, 134:169–189, 1997.

[35] T.Y. Hou, X. Wu, and Z. Cai. Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comput.*, 68:913–943, 1999.

[36] T. J. R. Hughes. Multiscale phenomena: Green's functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Computer Methods in Applied Mechanics and Engineering*, 127:387 – 401, 1995.

[37] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, and J-B Quincy. The variational multiscale method–a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166:3 – 24, 1998.

[38] J. Wang J and N. Zabaras. Hierarchical Bayesian models for inverse problems in heat conduction. *Inverse Problems*, 21(1):183–206, 2005.

[39] A. Jasra, A. Doucet, D.A. Stephens, and C.C. Holmes. Interacting sequential Monte Carlo samplers for trans-dimensional simulation. *Computational Statistics and Data Analysis*, 52(4):1765–1791, 2008.

[40] J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*. Springer, New York, 2005.

[41] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B*, 63(3):425–464, 2001.

[42] A. Klimke and B. Wohlmuth. Algorithm 847: Spinterp: piecewise multilinear hierarchical sparse grid interpolation in MATLAB. *ACM Transactions on Mathematical Software*, 31(4):561–579, 2005.

[43] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, IJCAI'95, pages 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[44] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[45] P.S. Koutsourelakis. A multi-resolution, non-parametric, Bayesian framework for identification of spatially-varying model parameters. *Journal of Computational Physics*, 228(17):6184 – 6211, 2009.

[46] F. R. Kschischang, B. J. Frey, and H-A Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transaction on Information Theory*, 47:498–519, 2001.

[47] S. Kullback. *Information Theory and Statistics*. John Wiley, 1959.

[48] S.L. Lauritzen. *Graphical Models*. Oxford Science Publications. Clarendon Press, 1996.

[49] H.K.H. Lee, D. Higdon, Z. Bi, M.A.R. Ferreira, and M. West. Markov random field models for high-dimensional parameters in simulations of fluid flow in porous media. Technical report, Technometrics, 2002.

[50] G. Li, C. Rosenthal, and H. Rabitz. High dimensional model representations. *The Journal of Physical Chemistry A*, 105(33):7765–7777, 2001.

[51] J.S. Liu. *Monte Carlo Strategies in Scientific Computing*. New York : Springer, 2008.

[52] J.S. Liu and R. Chen. Blind deconvolution via sequential imputations. *Journal of the American Statistical Association*, 90(430):567–576, 1995.

[53] J.S. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[54] X. Ma and N. Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.

[55] X. Ma and N. Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, May 2009.

[56] X. Ma and N. Zabaras. An efficient Bayesian inference approach to inverse problems based on an adaptive sparse grid collocation method. *Inverse Problems*, 25(3):035013, 2009.

[57] X. Ma and N. Zabaras. An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations. *Journal of Computational Physics*, 229(10):3884–3915, May 2010.

[58] X. Ma and N. Zabaras. A stochastic mixed finite element heterogeneous multiscale method for flow in porous media. *Journal of Computational Physics*, 230(12):4696–4722, June 2011.

[59] Y.M. Marzouk, H.N. Najm, and L.A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics*, 224(2):560 – 586, 2007.

[60] G. Mclachlan and D. Peel. *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley-Interscience, 1 edition, October 2000.

[61] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B*, 68(3):411 – 436, 2006.

[62] K.P. Murphy, Y. Weiss, and M.I. Jordan. Loopy belief propagation for approximate inference: an empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, UAI'99, pages 467–475, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[63] I.T. Nabney. *NETLAB: Algorithms for Pattern Recognition*. Advances in Pattern Recognition Series. Springer, 2004.

[64] R. B. Nelson. *An Introduction to Copulas (Lecture Notes in Statistics)*. Springer, 1998.

[65] F. Nobile, R. Tempone, and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309–2345, 2008.

[66] A. Nouy. Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Archives of Computational Methods in Engineering*, 16(3):251–285, 2009.

[67] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[68] F. Poirion and B. Puig. Simulation of non-gaussian multivariate stationary processes. *International Journal of Non-Linear Mechanics*, 45(5):587 – 597, 2010.

[69] R. Juanes R and F-X Dub. A locally conservative variational multiscale method for the simulation of porous media flow with multiscale source terms. *Computational Geosciences*, 12(3):273–295, 2008.

[70] P. Raviart. and J. Thomas. A mixed finite element method for 2-nd order elliptic problems. In Ilio Galligani and Enrico Magenes, editors, *Mathematical Aspects of Finite Element Methods*, volume 606 of *Lecture Notes in Mathematics*, pages 292–315. Springer Berlin / Heidelberg, 1977.

[71] N. Remy. S-GeMS: The Stanford Geostatistical Modeling Software: A Tool for New Algorithms Development. *Quantitative Geology and Geostatistics*, 14(4):865–871, 2005.

[72] S. Richardson and P.J. Green. On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society: Series B*, 59(4):731–792, 1997.

[73] G.G. Roger and D. Alireza. On the construction and analysis of stochastic models: Characterization and propagation of the errors associated with limited data. *Journal of Computational Physics*, 217:63 – 81, 2006.

[74] M. Rosenblatt. Remarks on a multivariate transformation. *Ann. Math. Statist.*, 23:470 – 472, 1952.

[75] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*, volume 104 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London, 2005.

[76] A. Sancetta and S. Satchell. The bernstein copula and its applications to

modeling and approximations of multivariate distributions. *Econometric Theory*, 20(3):pp. 535–562, 2004.

[77] K. Sargsyan, B. Debusschere, H. Najm, and O. Le Maître. Spectral representation and reduced order modeling of the dynamics of stochastic reaction networks via adaptive data partitioning. *SIAM Journal on Scientific Computing*, 31:4395–4421, 2010.

[78] M. Scutari. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35(3):1–22, 2010.

[79] S. Smolyak. Quadrature and interpolation formulas for tensor product of certain classes of function. *Soviet Mathematics Doklady*, 4:240–243, 1963.

[80] D. Sonjoy, G. Roger, and C.S. James. Asymptotic sampling distribution for polynomial chaos representation from data: A maximum entropy and fisher information approach. *SIAM Journal on Scientific Computing*, 30:2207–2234, 2008.

[81] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. The MIT Press, Cambridge, MA, USA, second edition, January 2001.

[82] H. Steck. Constraint-based structural learning in bayesian networks using finite data sets. 2001.

[83] E.B. Sudderth, A.T. Ihler, M. Isard, W.T. Freeman, and A.S. Willsky. Nonparametric belief propagation. *Commun. ACM*, 53(10):95–103, October 2010.

[84] A.N. Tikhonov. *Solution of Ill-Posed Problems*. Halster Press, Washington, 1985.

[85] A.B. Velamur and N. Zabaras. Stochastic inverse heat conduction using a spectral approach. *International Journal for Numerical Methods in Engineering*, 60:1569–1593, 2004.

[86] M.J. Wainwright and M.I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, January 2008.

[87] J. Wan and N. Zabaras. A Bayesian approach to multiscale inverse

problems using the sequential Monte Carlo method. *Inverse Problems*, 27(10):105004, October 2011.

[88] J. Wan and N. Zabaras. A bayesian approach to multiscale inverse problems using the sequential monte carlo method. *Inverse Problems*, 27(10):105004, 2011.

[89] J. Wan and N. Zabaras. A probabilistic graphical model approach to stochastic multiscale partial differential equations. *Journal of Computational Physics, in press*, 2013.

[90] J. Wan and N. Zabaras. Stochastic input model generation using bayesian network learning. *Journal of Computational Physics, to be submitted*, 2013.

[91] W.L. Wan, T.F. Chan, and B. Smith. An energy-minimizing interpolation for robust multigrid methods. *SIAM J. Sci. Comput.*, 21(4):1632–1649, 1999.

[92] X. Wan and G.E. Karniadakis. Solving elliptic problems with non-gaussian spatially-dependent random coefficients. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1985 – 1995, 2009.

[93] J. Wang and N. Zabaras. A Bayesian inference approach to the inverse heat conduction problem. *International Journal of Heat and Mass Transfer*, 47:3927–3941, 2004.

[94] J. Whittaker. *Graphical models in applied multivariate statistics*. Wiley series in probability and mathematical statistics: Probability and mathematical statistics. Wiley, 1990.

[95] D. Xiu. Efficient collocational approach for parametric uncertainty analysis. *Communications in Computational Physics*, 2(2):293–309, 2007.

[96] D. Xiu and J.S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118–1139, 2005.

[97] D. Xiu and G. E. Karniadakis. The Wiener–Askey Polynomial Chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24:619–644, 2002.

[98] D. Xiu and G. E. Karniadakis. Modeling uncertainty in flow simulations

via generalized polynomial chaos. *Journal of Computational Physics*, 187:137 – 167, 2003.

[99] X. Yang, M. Choi, G. Lin, and G.E. Karniadakis. Adaptive anova decomposition of stochastic incompressible and compressible flows. *Journal of Computational Physics*, 231(4):1587–1614, February 2012.

[100] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Generalized Belief Propagation. In *IN NIPS 13*, volume 13, pages 689–695, 2000.

[101] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. Technical Report TR-2001-22, Mitsubishi Electric Research Laboratories, January 2002.

[102] R. Yehezkel and B. Lerner. Bayesian Network Structure Learning by Recursive Autonomy Identification. *J. Mach. Learn. Res.*, 10:1527–1570, December 2009.